NIST Trustworthy and Responsible AI
NIST AI 100-2e2025

Adversarial Machine Learning

A Taxonomy and Terminology of Attacks and Mitigations

Apostol Vassilev
Alina Oprea
Alie Fordyce
Hyrum Anderson
Xander Davies
Maia Hamin

This publication is available free of charge from:
https://doi.org/10.6028/NIST.AI.100-2e2025

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

NIST Trustworthy and Responsible AI
NIST AI 100-2e2025

Adversarial Machine Learning

A Taxonomy and Terminology of Attacks and Mitigations

Apostol Vassilev
Computer Security Division
Information Technology Laboratory

Alina Oprea
Northeastern University

Maia Hamin
U.S. AI Safety Institute
National Institute of Standards and Technology

Alie Fordyce
Hyrum Anderson
Cisco

Xander Davies
U.K. AI Security Institute

This publication is available free of charge from:
https://doi.org/10.6028/NIST.AI.100-2e2025

March 2025

U.S. Department of Commerce
Howard Lutnick, Secretary

National Institute of Standards and Technology
Craig Burkhardt, Acting Under Secretary of Commerce for Standards and Technology and Acting NIST Director

NIST Technical Series Policies
Copyright, Use, and Licensing Statements
NIST Technical Series Publication Identifier Syntax

Publication History
Approved by the NIST Editorial Review Board on 2025-03-20

How to Cite this NIST Technical Series Publication:
Vassilev A, Oprea A, Fordyce A, Anderson H, Davies X, Hamin M (2025) Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations. (National Institute of Standards and Technology, Gaithersburg, MD) NIST Trustworthy and Responsible AI, NIST AI 100-2e2025.
https://doi.org/10.6028/NIST.AI.100-2e2025

Author ORCID iDs
Apostol Vassilev: 0000-0002-9081-3042
Alina Oprea: 0000-0002-4979-5292
Maia Hamin: 0009-0009-3834-6553

Contact Information
ai-100-2@nist.gov

Additional Information
Additional information about this publication is available at
https://csrc.nist.gov/pubs/ai/100/2/e2025/final, including related content, potential updates, and document history.

Abstract

This NIST Trustworthy and Responsible AI report provides a taxonomy of concepts and defines terminology in the field of adversarial machine learning (AML). The taxonomy is arranged in a conceptual hierarchy that includes key types of ML methods, life cycle stages of attack, and attacker goals, objectives, capabilities, and knowledge. This report also identifies current challenges in the life cycle of AI systems and describes corresponding methods for mitigating and managing the consequences of those attacks. The terminology used in this report is consistent with the literature on AML and is complemented by a glossary of key terms associated with the security of AI systems. Taken together, the taxonomy and terminology are meant to inform other standards and future practice guides for assessing and managing the security of AI systems by establishing a common language for the rapidly developing AML landscape.

Keywords

artificial intelligence; machine learning; attack taxonomy; abuse; data poisoning; evasion; privacy breach; attack mitigation; large language model; chatbot.

NIST Trustworthy and Responsible AI

The National Institute of Standards and Technology (NIST) promotes U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life. Among its broad range of activities, NIST contributes to the research, standards, evaluations, and data required to advance the development, use, and assurance of trustworthy artificial intelligence (AI).

Table of Contents

List of Figures

Audience

The intended primary audience for this document includes individuals and groups who are responsible for designing, developing, deploying, evaluating, and governing AI systems.

Background

This document is the result of an extensive literature review, conversations with experts in adversarial machine learning, and research performed by the authors in adversarial machine learning.

Trademark Information

All trademarks and registered trademarks belong to their respective organizations.

The Information Technology Laboratory (ITL) at NIST develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines.

This NIST Trustworthy and Responsible AI report focuses on identifying, addressing, and managing risks associated with adversarial machine learning. While practical guidance[1] published by NIST may serve as an informative reference, this guidance remains voluntary.

The content of this document reflects recommended practices. This document is not intended to serve as or supersede existing regulations, laws, or other mandatory guidance.

[1] In the context of this paper, the terms "practice guide," "guide," "guidance," and the like are consensus-created informative references that are intended for voluntary use. They should not be interpreted as equal to the use of the term "guidance" in a legal or regulatory context. This document does not establish any legal standard or any other legal requirement or defense under any law, nor does it have the force or effect of law.

How to Read This Document

This document uses the terms "AI technology," "AI system," and "AI applications" interchangeably. Terms related to the machine learning pipeline, such as "ML model" or "algorithm," are also used interchangeably in this document. Depending on context, the term "system" may refer to the broader organizational and/or social ecosystem within which the technology was designed, developed, deployed, and used instead of the more traditional use related to computational hardware or software.

Important reading notes:

- This document includes a series of blue callout boxes that highlight nuances and important takeaways.
- This document contains links shown in blue. Clicking on them will bring the reader to the relevant resource. Links links in the References point to external sources.
- Terms that are used but not defined or explained in the text are listed and defined in the Glossary. They are displayed in small caps in the text. Clicking on a word shown in SMALL CAPS (e.g., ADVERSARIAL EXAMPLE) takes the reader directly to the definition of that term in the Glossary. From there, one may click on the page number shown at the end of the definition to return.
- This document provides an Index of attack types to easily navigate and reference attacks and corresponding mitigations.

Acknowledgments

The authors wish to thank all of the people and organizations who submitted comments on the draft version of this paper. The received comments and suggested references were essential to improving the document and the future direction of this work. The authors also want to thank the many NIST, U.S. AI Safety Institute, and U.K. AI Security Institute colleagues who assisted in updating this document.

Author Contributions

The authors contributed equally to this work.

Predictive AI and Generative AI Taxonomy Index

- **Predictive AI Attacks Taxonomy**
  - Availability Violations (ID: NISTAML.01)
    - Model Poisoning (ID: NISTAML.011)
    - Clean-label Poisoning (ID: NISTAML.012)
    - Data Poisoning (ID: NISTAML.013)
    - Energy-latency (ID: NISTAML.014)
  - Integrity Violations (ID: NISTAML.02)
    - Clean-label Poisoning (ID: NISTAML.012)
    - Clean-label Backdoor (ID: NISTAML.021)
    - Evasion (ID: NISTAML.022)
    - Backdoor Poisoning (ID: NISTAML.023)
    - Targeted Poisoning (ID: NISTAML.024)
    - Black-box Evasion (ID: NISTAML.025)
    - Model Poisoning (ID: NISTAML.026)
  - Privacy Compromises (ID: NISTAML.03)
    - Model Extraction (ID: NISTAML.031)
    - Reconstruction (ID: NISTAML.032)
    - Membership Inference (ID: NISTAML.033)
    - Property Inference (ID: NISTAML.034)
  - Supply Chain Attacks (ID: NISTAML.05)
    - Model Poisoning (ID: NISTAML.051)
- **Generative AI Attacks Taxonomy**
  - Availability Violations (ID: NISTAML.01)
    - Data Poisoning (ID: NISTAML.013)
    - Indirect Prompt Injection (ID: NISTAML.015)
    - Prompt Injection (ID: NISTAML.018)
  - Integrity Violations (ID: NISTAML.02)

    - Data Poisoning (ID: NISTAML.013)
    - Indirect Prompt Injection (ID: NISTAML.015)
    - Prompt Injection (ID: NISTAML.018)
    - Backdoor Poisoning (ID: NISTAML.023)
    - Targeted Poisoning (ID: NISTAML.024)
    - Misaligned Outputs (ID: NISTAML.027)
- Privacy Compromises (ID: NISTAML.03)
    - Indirect Prompt Injection (ID: NISTAML.015)
    - Prompt Injection (ID: NISTAML.018)
    - Backdoor Poisoning (ID: NISTAML.023)
    - Membership Inference (ID: NISTAML.033)
    - Prompt Extraction (ID: NISTAML.035)
    - Leaking information from user interactions (ID: NISTAML.036)
    - Training Data Attacks (ID: NISTAML.037)
    - Data Extraction (ID: NISTAML.038)
    - Compromising connected resources (ID: NISTAML.039)
- Misuse Violations (ID: NISTAML.04)
    - Prompt Injection (ID: NISTAML.018)
- Supply Chain Attacks (ID: NISTAML.05)
    - Model Poisoning (ID: NISTAML.051)

Executive Summary

This NIST Trustworthy and Responsible AI report describes a taxonomy and terminology for ADVERSARIAL MACHINE LEARNING (AML) that may aid in securing applications of artificial intelligence (AI) against adversarial manipulations and attacks.

The statistical, data-based nature of ML systems opens up new potential vectors for attacks against these systems' security, privacy, and safety, beyond the threats faced by traditional software systems. These challenges span different phases of ML operations such as the potential for adversarial manipulation of training data; the provision of adversarial inputs to adversely affect the performance of the AI system; and even malicious manipulations, modifications, or interactions with models to exfiltrate sensitive information from the model's training data or to which the model has access. Such attacks have been demonstrated under real-world conditions, and their sophistication and impacts have been increasing steadily.

The field of AML is concerned with studying these attacks. It must consider the capabilities of attackers, the model or system properties that attackers might seek to violate in pursuit of their objectives, and the design of attack methods that exploit vulnerabilities during the development, training, and deployment phases of the ML life cycle. It is also concerned with the design of ML algorithms and systems that can withstand these security and privacy challenges, a property often known as robustness [274].

To taxonimize these attacks, this report differentiates between predictive and generative AI systems and the attacks relevant to each. It considers the components of an AI system including the data; the model itself; the processes for training, testing, and deploying the model; and the broader software and system contexts into which models may be embedded, such as cases where Generative Artificial Intelligence (GenAI) models are deployed with access to private data or equipped with tools to take actions with real-world consequences.

Thus, the attacks within this taxonomy are classified relative to: (i) the AI system type, (ii) the stage of the ML life cycle process in which the attack is mounted, (iii) the attacker's goals and objectives in terms of the system properties they seek to violate, (iv) the attacker's capabilities and access, and (v) the attacker's knowledge of the learning process and beyond.

This report adopts the concepts of security, resilience, and robustness of ML systems from the NIST AI Risk Management Framework. Security, resilience, and robustness are gauged by risk, which is a measure of the extent to which an entity (e.g., a system) is threatened by a potential circumstance or event (e.g., an attack) and the severity of the outcome should such an event occur. However, this report does not make recommendations on risk tolerance (i.e., the level of risk that is acceptable to organizations or society) because it is highly contextual and specific to applications and use cases.

The spectrum of effective attacks against ML is wide, rapidly evolving, and covers all phases of the ML lifecycle — from design and implementation to training, testing, and deployment in the real world. The nature and power of these attacks are different and their impacts may depend not only on the vulnerabilities of the ML models but also the weaknesses of the infrastructure in which the AI systems are deployed. AI system components may also be adversely affected by design and implementation flaws that cause failures outside the context of adversarial use, such as inaccuracy. However, these kinds of flaws are not within the scope of the literature on AML or the attacks in this report.

In addition to defining a taxonomy of attacks, this report provides corresponding methods for mitigating and managing the consequences of those attacks in the life cycle of AI systems, and outlines the limitations of widely used mitigation techniques to raise awareness and help organizations increase the efficacy of their AI risk-mitigation efforts. The terminology used in this report is consistent with the literature on AML and is complemented by a glossary that defines key terms associated with the field of AML in order to assist non-expert readers. Taken together, the taxonomy and terminology are meant to inform other standards and future practice guides for assessing and managing the security of AI systems by establishing a common language for the rapidly developing AML landscape. Like the taxonomy, the terminology and definitions are not intended to be exhaustive but rather to serve as a starting point for understanding and aligning on key concepts that have emerged in the AML literature.

1. Introduction

Artificial intelligence (AI) systems have been on a global expansion trajectory for several years [267]. These systems are being developed by and widely deployed into the economies of numerous countries, with increasing opportunities for people to use AI systems in many spheres of their lives [92]. This report distinguishes between two broad classes of AI systems: predictive AI (PredAI) and generative AI (GenAI). Although the majority of industrial applications of AI systems are still dominated by PredAI systems, there has been a recent increase in the adoption of GenAI systems in business and consumer contexts. As these systems permeate the digital economy and become essential parts of daily life, the need for their secure, robust, and resilient operation grows. These operational attributes are critical elements of trustworthy AI in the NIST AI Risk Management Framework [274] and the NCSC Machine Learning Principles [266].

The field of ADVERSARIAL MACHINE LEARNING (AML) studies attacks against ML systems that exploit the statistical, data-based nature of ML systems. Despite the significant progress of AI and machine learning (ML) in different application domains, these technologies remain vulnerable to attacks that can cause spectacular failures. The chances of these kinds of failure increase as ML systems are used in contexts where they may be subject to novel or adversarial interactions, and the consequences grow more dire as these systems are used in increasingly high-stakes domains. For example, in PredAI computer vision applications for object detection and classification, well-known cases of adversarial perturbations of input images have caused autonomous vehicles to swerve into lanes going in the opposite direction, stop signs to be misclassified as speed limit signs, and even people wearing glasses to be misidentified in high-security settings [121, 187, 332, 349]. Similarly, the potential for adversarial input to trick ML models into revealing hidden information has become more urgent as more ML models are being deployed in fields like medicine, where medical record leaks can expose sensitive personal information [25, 171].

In GenAI, large language models (LLMs) [13, 15, 49, 85, 102, 236, 247, 277, 279, 348, 365, 371, 372, 436] are increasingly becoming an integral part of software applications and internet infrastructure. LLMs are being used to create more powerful online search tools, help software developers write code, and power chatbots that are used by millions of people every day [255]. LLMs are also being augmented to create more useful AI systems, including through interactions with corporate databases and documents to enable powerful RETRIEVAL-AUGMENTED GENERATION (RAG) (RAG) [210] and through training- or inference-time techniques to enable LLMs to take real-world actions, such as browsing the web or using a bash terminal as an LLM-based AGENT [167, 261, 278, 419]. Thus, vulnerabilities in GenAI systems may expose a broad attack surface for threats to the privacy of sensitive user data or proprietary information about models' architecture or training data, and create risks to the integrity and availability of widely used systems.

As GenAI adoption has grown, the increasing capability of these systems has created another challenge for model developers: how to manage the risks created by unwanted or

harmful uses of these systems' capabilities.[275] As model developers have increasingly sought to apply technical interventions to reduce models' potential for misuse, another surface for high-stakes AML attacks has emerged in attacks that attempt to circumvent or disrupt these protections.

Fundamentally, many AI systems are susceptible both to AML attacks and to attacks that more closely resemble traditional cybersecurity attacks, including attacks against the platforms on which they are deployed. This report focuses on the former and considers the latter to be within the scope of traditional cybersecurity taxonomies.

Both PredAI and GenAI systems are vulnerable to attacks enabled by a range of attacker capabilities throughout the development and deployment life cycle. Attackers can manipulate training data [327], including the Internet data used in large-scale model training [57], or can modify test-time inference data and resources by adding adversarial perturbations or suffixes. Attackers can also attack the components used to make AI systems by inserting TROJAN functionality. As organizations increasingly rely on pre-trained models that could be used directly or fine-tuned with new datasets to enable different tasks, their vulnerability to these attacks increases.

Modern cryptography often relies on algorithms that are secure in an information-theoretic sense, that is, those that can be formally proven to ensure security under certain conditions. However, there are no information-theoretic security proofs for the widely used ML algorithms in modern AI systems. Moreover, information-theoretic *impossibility* results that set limits on the effectiveness of widely used mitigation techniques have begun to appear in the literature [124, 140, 432]. As a result, many of the advances in developing mitigations against different classes of AML attacks tend to be empirical and limited in nature, adopted because they appear to work in practice rather than because they provide information-theoretic security guarantees. Thus, many of these mitigations may themselves be vulnerable to new discoveries and evolutions in attacker techniques.

This report offers guidance for the development of:

- Standardized terminology for AML terms that can be used across relevant ML and cybersecurity communities. There are notable differences in terminology in different stakeholder communities and it is important to work towards bridging the differences as AI is increasingly adopted throughout enterprise and consumer contexts.
- A taxonomy of the most widely studied and currently effective attacks in AML, including:
  - Evasion, poisoning, and privacy attacks for PredAI systems
  - Poisoning, direct prompting, and indirect prompt injection attacks for GenAI systems
- A discussion of potential mitigations for these attacks and the limitations of existing mitigation techniques

NIST intends to update this report as new developments emerge in AML attacks and mitigations.

> This report provides a categorization of common classes of attacks and their mitigations for PredAI and GenAI systems. This report is not intended to provide an exhaustive survey of all available literature on Adversarial ML, which includes more than 11,354 references on arXiv.org since 2021 as of July 2024.

This report is organized into three sections.

- Section 2 considers PredAI systems. Section 2.1 introduces the taxonomy of attacks for PredAI systems, which defines the broad categories of attacker objectives and goals, and identifies the capabilities that an adversary must leverage to achieve the corresponding objectives. Specific attack classes are also introduced for each type of capability. Sections 2.2, 2.3, and 2.4 discuss the major classes of attacks: evasion, poisoning, and privacy, respectively. A corresponding set of mitigations for each class of attacks is provided in the attack class sections.
- Section 3 considers GenAI systems. Section 3.1 introduces the taxonomy of attacks for GenAI systems and defines the broad categories of attacker objectives and adversary capabilities relevant to these systems. Specific attack classes are introduced for each type of capability, along with relevant mitigations.
- Section 4 discusses remaining challenges in the field, including limitations to widely used mitigation techniques. The intent is to raise awareness of open questions in the field of AML and to call attention to trends that may shape risk and risk management practices in future.

2. Predictive AI Taxonomy

2.1. Attack Classification

Figure 1 introduces a taxonomy of attacks in AML on PredAI systems, based on attacker goals and objectives, capabilities, and knowledge.

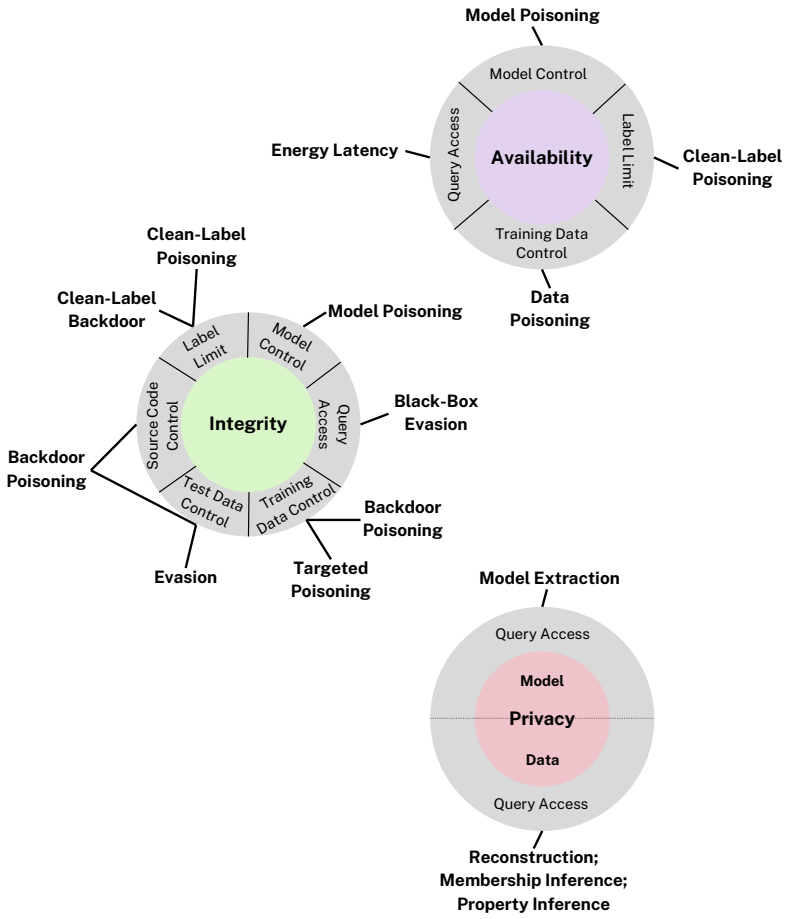

Figure 1. Taxonomy of attacks on PredAI systems

The attacker's objectives are shown as disjointed circles with the attacker's goal at the center of each circle: **availability** breakdown, **integrity** violation, and **privacy** compromise. The capabilities that an adversary must leverage to achieve their objectives are shown in the outer layer of the objective circles. Attack classes are shown as callouts connected to the capabilities required to mount each attack. Multiple attack classes that require the same capabilities to reach the same objective are shown in a single callout.

These attacks are classified according to the following dimensions: 1) learning method and stage of the learning process when the attack is mounted, 2) attacker goals and objectives,

3) attacker capabilities, and 4) attacker knowledge of the learning process. Several adversarial attack classification frameworks have been introduced in prior works [42, 358], and the goal here is to create a standard terminology for adversarial attacks on ML that unifies existing work.

2.1.1. Stages of Learning

Predictive machine learning involves a TRAINING STAGE in which a model is learned and a DEPLOYMENT STAGE in which the model is deployed on new, unlabeled data samples to generate predictions. In the case of SUPERVISED LEARNING, labeled training data is given as input to a training algorithm in the training stage, and the ML model is optimized to minimize a specific loss function. Validation and testing of the ML model is usually performed before the model is deployed in the real world. Common supervised learning techniques include CLASSIFICATION in which the predicted labels or *classes* are discrete and REGRESSION in which the predicted labels or *response variables* are continuous.

Other learning paradigms in the ML literature include UNSUPERVISED LEARNING, which trains models using unlabeled data at training time; SEMI-SUPERVISED LEARNING in which a small set of examples have labels, while the majority of samples are unlabeled; REINFORCEMENT LEARNING in which an agent interacts with an environment and learns an optimal policy to maximize its reward; FEDERATED LEARNING in which a set of clients jointly train an ML model by communicating with a server that performs an aggregation of model updates; and ENSEMBLE LEARNING, which is an approach that seeks better predictive performance by combining the predictions from multiple models.

Most PredAI models are DISCRIMINATIVE, i.e., learn only a decision boundary, such as LOGISTIC REGRESSION, SUPPORT VECTOR MACHINES, and CONVOLUTIONAL NEURAL NETWORKS. GenAI models may also be used in predictive tasks, such as sentiment analysis [125] .

AML literature predominantly considers adversarial attacks against AI systems that could occur at either the training stage or the deployment stage. During the training stage, the attacker might control part of the training data, their labels, the model parameters, or the code of ML algorithms, resulting in different types of poisoning attacks. During the deployment stage, the ML model is already trained, and the adversary could mount evasion attacks to create integrity violations and change the ML model's predictions, as well as privacy attacks to infer sensitive information about the training data or the ML model.

Training-time attacks. POISONING ATTACKS [40] occur during the ML training stage. In a DATA POISONING attack [40, 148], an adversary controls a subset of the training data by either inserting or modifying training samples. In a MODEL POISONING attack [222], the adversary controls the model and its parameters. Data poisoning attacks are applicable to all learning paradigms, while model poisoning attacks are most prevalent in federated learning [190], where clients send local model updates to the aggregating server, and in supply-chain attacks, where malicious code may be added to the model by suppliers of

model technology.

Deployment-time attacks. Other types of attacks can be mounted against deployed models. Evasion attacks modify testing samples to create ADVERSARIAL EXAMPLE [38, 144, 362], which are similar to the original sample (e.g., according to certain distance metrics) but alter the model predictions to the attacker's choices. Other attacks, such as availability attacks and privacy attacks including membership inference [342] and data reconstruction [110], can also be mounted by attackers with query access to a deployed ML model.

2.1.2. Attacker Goals and Objectives

The attacker's objectives are classified along three dimensions according to the three main types of security violations considered when analyzing the security of a system: availability breakdown, integrity violation, and privacy compromise. Figure 1 separates attacks into three disjointed circles according to their objective, and the attacker's objective is shown at the center of each circle.

Availability breakdown [NISTAML.01] [Back to Index]. An AVAILABILITY BREAKDOWN attack is a deliberate interference with a PredAI system to disrupt the ability of other users or processes to obtain timely and reliable access to its services. This attack type may be initiated at training or deployment time, although its impacts are typically experienced at deployment time. Availability attacks can be mounted via data poisoning, when the attacker controls a fraction of the training set; via model poisoning, when the attacker controls the model parameters; or as ENERGY-LATENCY ATTACK via query access. Data poisoning availability attacks have been proposed for SUPPORT VECTOR MACHINES [40], linear regression [179], and even neural networks [228, 260], while model poisoning attacks have been designed for neural networks [222] and federated learning [22].

- **Energy latency attacks [NISTAML.014] [Back to Index].** Recently, ENERGY-LATENCY ATTACK, a type of availability attacks that require only black-box access to the model, have been developed for neural networks across many different tasks in computer vision and natural language processing (NLP) [345].

Integrity violation [NISTAML.02] [Back to Index]. An INTEGRITY VIOLATION attack is a deliberate interference with a PredAI system to force it to misperform against its intended objectives and produce predictions that align with the adversary's objective. An attacker can cause an integrity violation by mounting an evasion attack at deployment time or a poisoning attack at training time. Evasion attacks require the modification of testing samples to create adversarial examples that are misclassified by the model while often remaining stealthy and imperceptible to humans [38, 144, 362]. Integrity attacks via poisoning can be classified as TARGETED POISONING ATTACK [137, 330], BACKDOOR POISONING ATTACK [148], and MODEL POISONING [22, 36, 123]. Targeted poisoning tries to violate the integrity of a few targeted samples and assumes that the attacker has training data control to insert the poisoned samples. Backdoor poisoning attacks require the generation of a BACKDOOR PATTERN,

which is added to both the poisoned samples and the testing samples to cause misclassification. Backdoor attacks are the only attacks in the literature that require both training and testing data control. Model poisoning attacks could result in either targeted or backdoor attacks, and the attacker modifies model parameters to cause an integrity violation. They have been designed for centralized learning [222] and federated learning [22, 36].

Privacy compromise [NISTAML.03] [Back to Index]. A PRIVACY COMPROMISE attack causes the unintended leakage of restricted or proprietary information from a PredAI system, including details about a model's training data, weights, or architecture [100, 309]. While the term "confidentiality" is more widely used in taxonomies of traditional cybersecurity attacks, the AML field has tended to use the top-level term "privacy" to encompass both attacks against the confidentiality of a model (e.g., those that extract information about a model's weights or architecture) and those that cause violations of expected privacy properties of model outputs (e.g. by exposing model training data) [310]. DATA CONFIDENTIALITY during ML training can be achieved through secure computation methods based on cryptographic techniques [2, 253, 288, 385], which ensure that training data and model parameters remain protected during the training phase. However, even models trained using paradigms that enforce data confidentiality may be vulnerable to privacy attacks, in which adversaries interacting with a model can extract information about its training data or parameters. In this report, we focus on privacy compromises that can occur at deployment time, regardless of the training method used, or whether data confidentiality was maintained during training.

In privacy attacks, attackers might be interested in learning information about the training data (resulting in DATA PRIVACY ATTACKS) or the ML model (resulting in MODEL PRIVACY ATTACKS). The attacker could have different objectives for compromising the privacy of training data, such as DATA RECONSTRUCTION [110] (inferring the content or features of training data), MEMBERSHIP-INFERENCE ATTACK [162, 343] (inferring the presence of data in the training set), TRAINING DATA EXTRACTION [59, 63] (extracting training data from generative models), ATTRIBUTE INFERENCE ATTACKS [184, 409] (inferring sensitive attributes of training records) and PROPERTY INFERENCE [134] (inferring properties about the training data distribution). MODEL EXTRACTION is a model privacy attack in which attackers aim to extract information about the model [177].

2.1.3. Attacker Capabilities

AML attacks for PredAI systems can be taxonomized with respect to the capabilities that an attacker controls. An adversary might leverage six types of capabilities to achieve their objectives, as shown in the outer layer of the objective circles in Fig. 1:

- TRAINING DATA CONTROL: The attacker might take control of a subset of the training data by inserting or modifying training samples. This capability is used in data poisoning attacks (e.g., availability poisoning, targeted or backdoor poisoning).

- MODEL CONTROL: The attacker might take control of the model parameters by either generating a Trojan trigger and inserting it in the model or by sending malicious local model updates in federated learning.
- TESTING DATA CONTROL: The attacker might add perturbations to testing samples at model deployment time, as performed in evasion attacks to generate adversarial examples or in backdoor poisoning attacks.
- LABEL LIMIT: This capability is relevant to restrict adversarial control over the labels of training samples in supervised learning. Clean-label poisoning attacks assume that the attacker does not control the label of the poisoned samples, while regular poisoning attacks assume label control over the poisoned samples.
- SOURCE CODE CONTROL: The attacker might modify the source code of the ML algorithm, such as the random number generator or any third-party libraries, which are often open source.
- QUERY ACCESS: The attacker might submit queries to the model and receive predictions (i.e., labels or model confidences), such as when interacting with an AI system hosted by a cloud provider as a machine learning as a service (MLaaS) offering. This capability is used by black-box evasion attacks, ENERGY-LATENCY ATTACK, and all privacy attacks that do not require knowledge of the model's training data, architecture, or parameters.

Even if an attacker does not have the ability to modify training/testing data, source code, or model parameters, access to these may still be crucial for mounting stronger white-box attacks that require knowledge of the ML system. See Sec. 2.1.4 for more details on attacker knowledge, and detailed definitions of white-box and black-box attacks.

Figure 1 connects each attack class with the capabilities required to mount the attack. For example, backdoor attacks that cause integrity violations require control of the training and testing data to insert the backdoor pattern. Backdoor attacks can also be mounted via source code control, particularly when training is outsourced to a more powerful entity. Clean-label backdoor attacks do not allow label control on the poisoned samples in addition to the capabilities needed for backdoor attacks.

2.1.4. Attacker Knowledge

Another dimension of attack classification is how much knowledge the attacker has about the ML system. There are three main types of attacks:

White-box attacks. These assume that the attacker operates with *full* knowledge about the ML system, including the training data, model architecture, and model hyperparameters. While these attacks operate under very strong assumptions, the main reason for analyzing them is to test the vulnerability of a system against worst-case adversaries and

to evaluate potential mitigations. This definition is more general and encompasses the notion of adaptive attacks in which knowledge of the mitigations applied to the model or the system is explicitly tracked.

Black-box attacks. These attacks assume that the attacker operates with minimal, and sometimes no knowledge at all about the ML system. An adversary might have query access to the model, but they have no other information about how the model is trained. These attacks are the most practical since they assume that the attacker has no knowledge of the AI system and utilizes system interfaces readily available for normal use.

Gray-box attacks. There are a range of gray-box attacks that capture adversarial knowledge between black-box and white-box attacks. Suciu et al. [358] introduced a framework to classify gray-box attacks. An attacker might know the model architecture but not its parameters, or the attacker might know the model and its parameters but not the training data. Other common assumptions for gray-box attacks are that the attacker has access to data distributed identically to the training data and knows the feature representation. The latter assumption is important for applications in which feature extraction is used before training an ML model, such as cybersecurity, finance, and healthcare.

2.1.5. Data Modality

Until recently, most attacks and defenses in adversarial machine learning have operated under a single modality, but a new trend in the field is to use multimodal data. The taxonomy of attacks defined in Fig. 1 is independent of the modality of the data in specific applications.

The most common data modalities in the AML literature include:

- **Image:** Adversarial examples of image data [144, 362] have the advantage of a continuous domain, and gradient-based methods can be applied directly for optimization. Backdoor poisoning attacks were first invented for images [148], and many privacy attacks are run on image datasets (e.g., [342]). The image modality includes other types of imaging (e.g., LIDAR, SAR, IR, hyperspectral).
- **Text:** Text is a popular modality, and all classes of attacks have been proposed for text models, including evasion [150], poisoning [82, 213], and privacy [426].
- **Audio:** Audio systems and text generated from audio signals have also been attacked [66].
- **Video:** Video comprehension models have shown increasing capabilities in vision and language tasks [428], but such models are also vulnerable to attacks [402].
- **Cybersecurity**[2]**:** The first poisoning attacks were discovered in cybersecurity for worm

[2]Cybersecurity data may not include a single modality but rather multiple modalities, such as network-level, host-level, or program-level data.

signature generation (2006) [291] and spam email classification (2008) [269]. Since then, poisoning attacks have been shown for malware classification, malicious PDF detection, and Android malicious app classification [329]. Evasion attacks against similar data modalities have been proposed as well: malware classification [103, 357], PDF malware classification [352, 414], Android malicious app detection [295], and network instrusion detection [93]. Poisoning unsupervised learning models has been shown for clustering used in malware classification [41] and network traffic anomaly detection [315].

Anomaly detection based on data-centric approaches allows for automated feature learning through ML algorithms. However, the application of ML to such problems comes with specific challenges related to the need for very low false negative and low false positive rates (e.g., the ability to catch zero-day attacks). This challenge is compounded by the fact that trying to accommodate all of these together makes ML models susceptible to adversarial attacks [198, 301, 446].

- **Tabular data:** There have been numerous attacks against ML models working on tabular data, such as poisoning availability attacks against healthcare and business applications [179], privacy attacks against healthcare data [422], and evasion attacks against financial applications [141].

Recently, the use of ML models trained on multimodal data has gained traction, particularly the combination of image and text data modalities. Several papers have shown that multimodal models may provide some resilience against attacks [417], but other papers show that multimodal models themselves could be vulnerable to attacks mounted on all modalities at the same time [77, 333, 415] (see Sec. 4.2.3).

An open challenge is to test and characterize the resilience of a variety of multimodal ML models against evasion, poisoning, and privacy attacks.

2.2. Evasion Attacks and Mitigations

[**NISTAML.022**] [Back to Index]

The discovery of evasion attacks against ML models has led to significant growth in AML research over the last decade. In an evasion attack, the adversary's goal is to generate adversarial examples: samples whose classification can be changed to an arbitrary class of the attacker's choice – often with only minimal perturbation [362]. For example, in the context of image classification, the perturbation of the original sample might be small so that a human cannot observe the transformation of the input; while the ML model can be tricked to classify the adversarial example in the target class selected by the attacker, humans still recognize it as part of the original class.

Early known instances of evasion attacks date back to 1988 with the work of Kearns and Li [192] and 2004 when Dalvi et al. [98] and Lowd and Meek [226] demonstrated the existence of adversarial examples for linear classifiers used in spam filters. Later, Szedegy et al. [362] showed that deep neural networks used for image classification could be easily manipulated through adversarial examples. In 2013, Szedegy et al. [362] and Biggio et al. [38] independently discovered an effective method for generating adversarial examples against linear models and neural networks by applying gradient optimization to an adversarial objective function. Both of these techniques require white-box access to the model and were improved by subsequent methods that generated adversarial examples with even smaller perturbations [20, 65, 232].

Adversarial examples are also applicable in more realistic black-box settings in which attackers only obtain query access capabilities to the trained model. Even in the more challenging black-box setting in which attackers obtain the model's predicted labels or confidence scores, deep neural networks are still vulnerable to adversarial examples. Methods for creating adversarial examples in black-box settings include zeroth-order optimization [80], discrete optimization [254], Bayesian optimization [344], and *transferability*, which involves the white-box generation of adversarial examples on a different model before transferring them to the target model [282, 283, 377]. While cybersecurity and image classifications were the first application domains to showcase evasion attacks, ML technology in many other application domains has come under scrutiny, including speech recognition [66], natural language processing [185], and video classification [215, 401].

Mitigating adversarial examples is a well-known challenge in the community and deserves additional research and investigation. The field has a history of publishing defenses evaluated under relatively weak adversarial models that are subsequently broken by more powerful attacks. Mitigations need to be evaluated against strong adaptive attacks, and guidelines for the rigorous evaluation of newly proposed mitigation techniques have been established [97, 375]. The most promising directions for mitigating the critical threat of evasion attacks are adversarial training [144, 232] (iteratively generating and inserting adversarial examples with their correct labels at training time); certified techniques, such as

randomized smoothing [94] (evaluating ML prediction under noise); and formal verification techniques [136, 191] (applying formal method techniques to verify the model's output). Nevertheless, these methods have different limitations, such as decreased accuracy for adversarial training and randomized smoothing and computational complexity for formal methods. There is an inherent trade-off between robustness and accuracy [374, 379, 433]. Similarly, there are trade-offs between a model's robustness and fairness guarantees [71].

2.2.1. White-Box Evasion Attacks

In the white-box threat model, the attacker has full knowledge of the model architecture and parameters, as discussed in Section 2.1.4. The main challenge for creating adversarial examples in this setting is to find a perturbation added to a testing sample that changes its classification label, often with constraints on properties such as the perceptibility or size of the perturbation. In the white-box threat model, it is common to craft adversarial examples by solving an optimization problem written from the attacker's perspective, which specifies the objective function for the optimization (such as changing the target label to a certain class), as well as a distance metric to measure the similarity between the testing sample and the adversarial example.

Optimization-based methods. Szedegy et al. [362] and Biggio et al. [38] independently proposed the use of optimization techniques to generate adversarial examples. In their threat models, the adversary is allowed to inspect the entirety of the ML model and compute gradients relative to the model's loss function. These attacks can be targeted (i.e., the adversarial example's class is selected by the attacker) or untargeted (i.e., the adversarial examples are misclassified to any other incorrect class).

Szedegy et al. [362] coined the widely used term *adversarial examples*. They considered an objective that minimized the ℓ_2 norm of the perturbation subject to the model prediction changing to the target class. The optimization is solved using the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) method. Biggio et al. [38] considered the setting of a binary classifier with malicious and benign classes with a continuous and differentiable discriminant function. The objective of the optimization is to minimize the discriminant function in order to generate adversarial examples of maximum confidence.

While Biggio et al. [38] applied their method to linear classifiers, kernel SVM, and multilayer perceptrons, Szedegy et al. [362] showed the existence of adversarial examples on deep learning models used for image classification. Goodfellow et al. [144] introduced an efficient method for generating adversarial examples for deep learning: the Fast Gradient Sign Method (FGSM), which performs a single iteration of gradient descent for solving the optimization. This method has been extended to an iterative FGSM attack by Kurakin et al. [200].

Subsequent works have proposed new objectives and methods for optimizing the generation of adversarial examples with the goals of minimizing the perturbations and supporting