Tracing the Adoption of Adversarial Machine Learning Research into Industry Practice (2014–2025)

[Author Names] Affiliations>Affiliations

[emails]

Abstract

Despite a decade of adversarial machine learning (AML) research producing over 66 catalogued attack techniques and numerous defense mechanisms, industry adoption remains limited—surveys indicate only 5% of practitioners have experienced AI-specific attacks, yet 86% express security concerns. This systematic review measures the temporal lag between publication of AML research and evidence of industry adoption across tool integration, benchmark inclusion, regulatory citation, and vendor acknowledgment. Using an artifact-anchored methodology, we trace techniques implemented in major security tools (IBM ART, CleverHans, PyRIT), standardized benchmarks (RobustBench, HarmBench), and regulatory frameworks (MITRE ATLAS, OWASP Top 10) back to their originating academic papers. We code approximately 100 papers spanning 2014–2025 across foundational attacks, privacy threats, physical-world attacks, and LLM-specific vulnerabilities. We examine how adoption speed varies across domains and identify acceleration factors including standardized evaluation, regulatory mandates, and commercial investment. Our coding framework and adoption evidence database provide a foundation for future research-practice alignment studies.

1 Introduction

Machine learning systems now influence decisions affecting millions daily—from facial recognition at border crossings [Grother et al., 2019] to fraud detection in financial services [Dal Pozzolo et al., 2015] and content moderation on social platforms [Gorwa et al., 2020]. This widespread deployment has intensified scrutiny of adversarial vulnerabilities: inputs or interactions crafted to cause models to behave in unintended ways [Biggio and Roli, 2018].

The field of adversarial machine learning emerged with Szegedy et al.'s demonstration that imperceptible perturbations could fool state-of-the-art classifiers [Szegedy et al., 2014], followed by Goodfellow et al.'s Fast Gradient Sign Method establishing the dominant attack paradigm [Goodfellow et al., 2015]. Over the subsequent decade, researchers catalogued 66 attack techniques spanning evasion, poisoning, and privacy threats [MITRE Corporation, 2024]. MITRE ATLAS now documents 33 real-world case studies, and regulatory frameworks including the EU AI Act mandate adversarial robustness testing for high-risk systems [European Parliament and Council, 2024].

Yet industry surveys reveal a persistent gap. Kumar et al. [Kumar et al., 2020] found practitioners "not equipped with tactical and strategic tools" for ML-specific attacks. Grosse et al. [Grosse et al., 2023] reported only 5% of AI practitioners had experienced AI-specific attacks, despite 86% expressing concern. Mink et al. [Mink et al., 2023] identified organizational barriers including lack of institutional motivation, inability to assess AML risk, and structures discouraging implementation. Apruzzese et al. [Apruzzese et al., 2023] crystallized these concerns, arguing that "real attackers don't compute gradients"—academic threat models assume capabilities rarely available in practice.

This gap matters because real-world incidents demonstrate adversarial threats are not merely theoretical. In November 2025, Anthropic disclosed the first documented large-scale AI-orchestrated cyber campaign, with Chinese state-sponsored actors using Claude Code to execute 80–90% of operational tasks autonomously [Anthropic, 2025]. Tesla Autopilot was fooled by adversarial tape modifications [Tencent Keen Security Lab, 2019, McAfee Advanced Threat Research, 2020]. Training data extraction from ChatGPT recovered megabytes of verbatim training data for under $200 [Nasr et al., 2023]. The OWASP Top 10 for LLM Applications lists prompt injection as the #1 vulnerability across all versions [OWASP Foundation, 2024].

1.1 Research Questions

We address three questions about the research-to-practice transfer in adversarial ML:

1. **RQ1 (Adoption Lag):** What is the typical time lag between publication of landmark AML research and evidence of industry adoption, measured through tool integration, commercial reference, regulatory citation, and production deployment?
2. **RQ2 (Domain Variation):** How does adoption speed vary across application domains (computer vision, NLP, malware detection, autonomous systems, LLMs), and what factors explain these differences?
3. **RQ3 (Acceleration Factors):** What mechanisms—regulatory frameworks, standardized benchmarks, industry consortiums, commercial tools—have accelerated adoption, particularly for foundation model security post-2022?

1.2 Contributions

This review makes three contributions:

1. **Artifact-anchored methodology:** We introduce a reverse-engineering approach that traces industry artifacts back to source papers, ensuring every paper in our sample has verified adoption evidence by construction.
2. **Reproducible coding framework:** We release a 12-variable codebook (Table 5) with decision rules for coding paper characteristics and measuring adoption timelines.
3. **Quantified adoption analysis:** We provide the first systematic measurement of research-to-industry lag across domains (vision, NLP, LLM, malware), artifact types, and publication eras.

2 Background

2.1 The Emergence of Adversarial Vulnerabilities

Modern adversarial ML research began with Szegedy et al.'s December 2013 demonstration that small perturbations could cause misclassification with high confidence [Szegedy et al., 2014]. This work, which received ICLR's 2024 Test of Time Award, revealed that adversarial examples transfer across independently trained models—a finding with profound security implications.

Goodfellow et al. [Goodfellow et al., 2015] attributed these vulnerabilities to linear behavior in high-dimensional spaces and introduced the Fast Gradient Sign Method (FGSM), appearing on arXiv December 20, 2014 and published at ICLR 2015. FGSM established gradient-based perturbation as the dominant paradigm and introduced adversarial training as a defense. These foundational works established conventions that shaped subsequent research: white-box access assumptions, L_p perturbation constraints, and optimization-based attack formulations.

Subsequent work refined attack formulations. The Carlini & Wagner attack [Carlini and Wagner, 2017] (IEEE S&P 2017) achieved state-of-the-art success across multiple norms. Madry et al.'s PGD attack [Madry et al., 2018] (ICLR 2018) established the robust optimization framework:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim D} \left[\max_{\|\delta\|\leq\epsilon} L(f_\theta(x+\delta), y) \right] \tag{1}$$

This min-max formulation remains the gold standard—10 of the top-10 models on RobustBench use PGD-based adversarial training [Croce et al., 2021].

2.2 Expanding Threat Landscape

Research expanded beyond test-time evasion to encompass the full ML pipeline. **Privacy attacks** demonstrated that models leak information: Shokri et al. [Shokri et al., 2017] introduced membership inference at IEEE S&P 2017; Tramèr et al. [Tramèr et al., 2016] demonstrated model extraction from commercial APIs at USENIX Security 2016; Carlini et al. [Carlini et al., 2021] extracted verbatim training data from GPT-2 at USENIX Security 2021.

Integrity attacks compromise models during training. BadNets [Gu et al., 2017] (arXiv August 2017) demonstrated backdoor injection through poisoned training data. Subsequent work extended backdoors to federated learning [Bagdasaryan et al., 2020] and self-supervised learning [Jia et al., 2021].

Physical-world attacks demonstrated that adversarial examples survive real-world conditions. Eykholt et al. [Eykholt et al., 2018] (CVPR 2018) created adversarial stop signs robust to viewing angles and distances. DolphinAttack [Zhang et al., 2017] (ACM CCS 2017, Best Paper) compromised voice assistants via ultrasonic commands inaudible to humans.

2.3 The LLM Security Paradigm Shift

Large language models introduced qualitatively different adversarial challenges. Unlike traditional attacks requiring gradient access and imperceptible perturbations, LLM attacks exploit semantic properties through natural language.

Prompt injection was first documented by Simon Willison in September 2022 and formalized by Perez & Ribeiro [Perez and Ribeiro, 2022] at the NeurIPS 2022 Workshop on ML Safety. Greshake et al. [Greshake et al., 2023] demonstrated indirect prompt injection against retrieval-augmented systems (arXiv February 2023, ACM AISec November 2023), showing attackers can embed malicious instructions in external content.

Automated jailbreaking emerged with Zou et al.'s GCG attack [Zou et al., 2023] (arXiv July 2023, NeurIPS 2023 Spotlight), which optimizes adversarial suffixes achieving the first automated jailbreaks against aligned LLMs including ChatGPT, Bard, and Claude. Subsequent work developed more efficient black-box methods: PAIR [Chao et al., 2024] achieves jailbreaks in approximately 20 queries; TAP [Mehrotra et al., 2024] uses tree-of-thought reasoning for further efficiency.

LLM defenses include alignment techniques (RLHF [Ouyang et al., 2022], Constitutional AI [Bai et al., 2022]), specialized guardrails (LlamaGuard [Inan et al., 2023] released December 2023, NeMo Guardrails open-sourced April 2023), and input/output filtering. However, these defenses exhibit brittleness—the HackAPrompt competition [Schulhoff et al., 2024] saw all 44 defenses eventually bypassed across 137,000+ adversarial interactions.

2.4 The Theory-Practice Gap

Apruzzese et al. [Apruzzese et al., 2023] synthesized concerns about research-practice disconnect through real-world case studies at IEEE SaTML 2023. Their core observation: academic threat models assume attackers possess white-box access, unlimited queries, and gradient computation capabilities rarely available in practice. Real incidents typically involve simpler tactics—basic input manipulation, system-level exploitation, social engineering.

Industry surveys quantify this gap. Kumar et al. [Kumar et al., 2020] interviewed 28 organizations at IEEE S&P Workshops 2020, finding widespread uncertainty about assessing adversarial risks. Grosse et al. [Grosse et al., 2023] surveyed 139 practitioners (IEEE TIFS 2023), finding only 5% had experienced AI-specific attacks despite 86% expressing concern. Mink et al. [Mink et al., 2023] conducted 21 interviews at USENIX Security 2023, identifying three barriers: lack of institutional motivation, inability to assess AML risk, and organizational structures discouraging implementation.

However, prior work characterized this gap qualitatively. Our contribution is to measure adoption timelines quantitatively, identifying specific lag durations and acceleration factors. This work does not assess the severity of adversarial threats, but quantitatively measures when research techniques enter practitioner ecosystems.

3 Methodology

3.1 Study Design Overview

This study measures the temporal lag between adversarial machine learning (AML) research and demonstrable industry adoption. Rather than starting from academic publications and speculating about downstream impact, we adopt an *artifact-anchored backward traceability* methodology. We begin with concrete industry artifacts—open-source security tools, standardized benchmarks, regulatory frameworks, and vendor security documentation—and trace each adopted technique back to the academic paper that originally introduced it.

This design ensures that every paper in our dataset has verifiable evidence of adoption by construction, enabling precise measurement of research-to-practice timelines while avoiding subjective judgments about a paper's "importance" or speculative claims about industry relevance.

3.2 Artifact Universe Definition

Before extracting any papers, we define and freeze the set of industry artifacts examined in this study. Artifacts were selected according to objective inclusion criteria to minimize selection bias.

3.2.1 Open-Source Security Tools

We include open-source AML tools satisfying the criterion of $\geq$1,000 GitHub stars, ensuring broad community adoption and active maintenance. Based on this criterion, we analyze five tools (Table 1):

Table 1: Selected open-source AML tools

Tool	Stars	Description
CleverHans	6,401	First major AML library (University of Toronto)
IBM ART	5,789	Adversarial Robustness Toolbox (Linux Foundation)
TextAttack	3,348	NLP adversarial attacks
PyRIT	3,343	LLM red-teaming toolkit (Microsoft)
Foolbox	2,936	Adversarial attacks (Bethge Lab)

For each tool, we clone the Git repository and programmatically extract arXiv references from source code, documentation, docstrings, and bibliography files. Adoption dates are determined via `git log -follow` to identify the first commit referencing each paper.

3.2.2 Standardized Benchmarks

We include benchmarks that are (1) published in peer-reviewed venues and (2) evaluate AML attacks or defenses as named techniques with explicit paper citations:

Table 2: Selected AML benchmarks

Benchmark	Venue	Description
RobustBench	NeurIPS 2021	Adversarial robustness leaderboard (120+ models)
AutoAttack	ICML 2020	Standardized attack ensemble evaluation
HarmBench	ICML 2024	LLM jailbreak evaluation (33 LLMs, 18 methods)

Only techniques explicitly evaluated and attributed to specific academic papers are included. Adoption dates are extracted from Git commit history of benchmark repositories.

3.2.3 Regulatory and Threat Frameworks

We include MITRE ATLAS [MITRE Corporation, 2024], the industry-standard adversarial ML threat framework modeled after ATT&CK. ATLAS catalogs 66 adversarial ML techniques with explicit academic references, providing structured machine-readable data via its public GitHub repository (`atlas-data`). We programmatically extract technique IDs, names, and cited papers from this repository.

3.2.4 Artifact Summary

Table 3 summarizes the nine Git-searchable artifacts included in our automated extraction pipeline, along with their selection criteria.

Table 3: Summary of artifacts and selection criteria

Category	Criterion	Artifacts
Tools	$\geq$1,000 GitHub stars	CleverHans, IBM ART, TextAttack, PyRIT, Foolbox
Benchmarks	Peer-reviewed publication	RobustBench, AutoAttack, HarmBench
Regulatory	Industry threat framework	MITRE ATLAS

3.3 Automated Paper Extraction

For each of the nine Git-searchable artifacts, we clone the repository and programmatically extract arXiv references using comprehensive regular expression patterns that capture:

- Direct arXiv URLs (`arxiv.org/abs/`, `arxiv.org/pdf/`)
- BibTeX `eprint` fields
- Inline citations (e.g., “arXiv:2401.12345”)

We scan all relevant file types: Python source (`.py`), documentation (`.md`, `.rst`, `.txt`), configuration (`.json`, `.yaml`, `.yml`), and bibliography (`.bib`) files. Each extracted arXiv ID is resolved to a canonical paper record via the arXiv API, retrieving title, authors, and publication date.

Exclusions: We exclude non-AML papers (e.g., foundational ML papers like Adam, ResNet, VGGNet) based on a predefined exclusion list validated against paper abstracts.

3.4 Paper Selection Criteria

From the full set of 277 AML papers extracted from the nine artifacts, we select papers for manual coding using a tiered approach:

1. **Multi-artifact papers (61 papers):** Papers cited by ≥2 artifacts provide the strongest evidence of cross-ecosystem adoption.
2. **MITRE ATLAS-only papers (10 papers):** Papers cited exclusively by MITRE ATLAS represent regulatory adoption without tool/benchmark integration, capturing a distinct adoption pathway.

This yields a final sample of **71 papers** for manual coding, balancing methodological rigor (multi-artifact validation) with coverage of regulatory adoption patterns.

3.5 Paper Validation

Each extracted reference is validated to ensure: (1) the paper introduces the technique (not merely applies or evaluates it), (2) the technique name corresponds to the artifact reference, and (3) the paper is publicly accessible via arXiv or DOI.

3.6 Adoption Event Definition

We define three observable adoption events, corresponding to our artifact categories:

Table 4: Adoption event definitions

Adoption Event	**Operational Definition**
Tool adoption	First Git commit referencing the paper
Benchmark adoption	Paper cited in benchmark repository
Regulatory adoption	Paper referenced in MITRE ATLAS

Adoption is defined strictly as acknowledgment or integration, not mitigation or successful defense.

3.7 Adoption Lag Measurement

For each (paper, adoption event) pair, we compute:

$$\text{Adoption Lag} = \text{Date}_{\text{artifact}} - \text{Date}_{\text{publication}} \tag{2}$$

Artifact dates are defined as:

- **Tools:** First Git commit timestamp referencing the paper (UTC)
- **Benchmarks:** First Git commit timestamp referencing the paper (UTC)
- **Regulatory:** First Git commit timestamp in MITRE ATLAS repository (UTC)

Publication dates are defined as:

- **Peer-reviewed:** Conference/journal publication date
- **arXiv-first:** First arXiv submission date

For papers cited by multiple artifacts, we record all adoption events and use the *earliest* adoption date for computing first adoption lag. This captures when a paper first entered the practitioner ecosystem, regardless of which artifact adopted it first.

3.8 Paper Coding Scheme

Each paper is coded along three dimensions using a fixed codebook (Table 5). All coding is performed by the authors based on the paper text only. Threat model attributes are coded using author-stated assumptions; if unstated, code as "Not specified."

Table 5: Complete codebook for paper coding

Variable	**Values**	**Coding Rule**
Research Characteristics (G1–G7)		
G1: Type	Attack / Defense / Evaluation	Primary contribution
G2: Threat category	Evasion / Poisoning / Privacy / N/A	Attack category; N/A for defenses
G3: Domain	Vision / NLP / Malware / Audio / Tabular / LLM / Cross-domain	Primary evaluation domain
G4: Venue	ML / Security / Journal / arXiv-only	See venue list below
G5: Code available	Yes / No	Code link exists at time of coding
G6: Code timing	At-pub / Post-pub / Never	At-pub = within 1 month of paper
G7: Year	2014–2025	Earliest of arXiv or venue date
Threat Model (T1–T2) – Attack papers only		
T1: Access level	White / Gray / Black	White = full model access
T2: Gradient required	Yes / No	Gradients used at any stage
Practical Evaluation (Q1–Q3)		
Q1: Real-world eval	Yes / Partial / No	Yes = production system tested
Q2: Cost reported	Yes / No	Explicit FLOPs, time, or queries
Q3: Defense-aware	Yes / No / N/A	N/A for defense papers

Venue classification: ML = NeurIPS, ICML, ICLR, CVPR, ECCV, ICCV, ACL, EMNLP, NAACL. Security = IEEE S&P, ACM CCS, USENIX Security, NDSS, IEEE SaTML. Journal = TPAMI, TIFS, TDSC, etc. arXiv-only = no peer-reviewed venue.

3.8.1 Decision Rules

1. **G1:** If paper proposes both attack and defense, code based on which receives more experimental evaluation.
2. **G2:** For defense papers, code "N/A." Evasion = test-time input perturbation. Poisoning = training-time data manipulation. Privacy = membership inference, model extraction, training data extraction.
3. **G3:** Code "Cross-domain" only if paper evaluates on 2+ distinct domains with separate experiments.

4. **G4:** Use venue of first peer-reviewed publication. If workshop paper later becomes full paper, use full paper venue.
5. **T1/T2:** For defense papers, leave blank (these apply to attacks only). White-box = gradients from target model. Gray-box = surrogate model gradients transferred to target. Black-box = query access only, no gradients.
6. **Q1:** "Yes" = tested on production system (commercial API, deployed vehicle). "Partial" = realistic simulation or industry dataset. "No" = standard benchmarks only (CIFAR, ImageNet, etc.).
7. **Q3:** For defense papers, code "N/A." For attacks, "Yes" = tests against adaptive defenses or AutoAttack.

Coding procedure: For each paper: (1) record metadata (title, authors, DOI/arXiv, publication date, venue); (2) read abstract and introduction to identify technique name and G1 (type); (3) locate threat model section to code T1 and T2 (attack papers only); (4) review experiments to code G3, Q1, Q2, Q3; (5) check code availability for G5, G6; (6) record adoption events with dates from artifacts; (7) calculate adoption lag in months. Estimated time: 20–30 minutes per paper.

3.9 Statistical Analysis

We address each research question through pre-specified analyses:

RQ1 (Adoption Lag): We report distributions of first-adoption lag, stratified by artifact type (tool, benchmark, regulatory). We compare lags across publication eras (2014–2017, 2018–2021, 2022–2025) using Kruskal-Wallis tests with post-hoc Dunn tests.

RQ2 (Domain Variation): We compare adoption lags across domains using pairwise Mann-Whitney U tests with Bonferroni correction ($\alpha = 0.05/\binom{n}{2}$ for n domains). Primary hypothesis: LLM security papers show significantly shorter lags than computer vision papers.

RQ3 (Acceleration Factors): We fit a Cox proportional hazards model predicting time-to-first-adoption, with covariates: publication year (continuous), domain (categorical), venue type (ML vs. Security), code availability (binary), and threat model (white-box vs. gray-box vs. black-box). Hazard ratios >1 indicate faster adoption. Model diagnostics include proportional hazards tests and residual analysis.

3.10 Reliability, Reproducibility, and Limitations

Reliability: We employ intra-rater reliability (15% of papers re-coded after two weeks, targeting $\kappa \geq 0.80$) and inter-rater reliability for papers coded by multiple authors. Adoption dates are verified via `git log` and Wayback Machine archives.

Data release: All extracted metadata, coding decisions, and artifact URLs will be released as a structured CSV dataset with accompanying codebook.

Limitations: (1) By construction, our sample excludes papers never adopted—we measure adoption timelines, not adoption rates. (2) Our nine Git-searchable artifacts may miss proprietary implementations; adoption timelines represent lower bounds. (3) Git commit dates represent code-level adoption, which may lag behind internal awareness. (4) Observational design precludes causal claims. (5) We focus on English-language publications and arXiv-indexed papers.

4 Artifact Timeline

Table 6 documents the nine artifacts included in our automated extraction, with first public release dates verified from GitHub and official announcements. These dates establish the timeline against which we measure research adoption.

Table 6: Artifact release timeline

Category	Artifact	First Release	Stars/Venue
Tools	CleverHans	Oct 2016	6,401 stars
	Foolbox	Jul 2017	2,936 stars
	IBM ART	Jul 2018	5,789 stars
	TextAttack	May 2020	3,348 stars
	PyRIT	Feb 2024	3,343 stars
Benchmarks	AutoAttack	Mar 2020	ICML 2020
	RobustBench	Oct 2020	NeurIPS 2021
	HarmBench	Feb 2024	ICML 2024
Regulatory	MITRE ATLAS	Jun 2021	Industry framework

5 Results

[TODO: Complete after paper coding. This section will report:

- Sample construction summary (extraction counts by artifact source)
- RQ1: Adoption lag distributions (overall, by artifact type, by publication era)
- RQ2: Domain variation analysis (lag comparisons across Vision, NLP, LLM, Malware, Audio)
- RQ3: Cox regression results (predictors of adoption speed)

]

6 Discussion

[TODO: Complete after results analysis. This section will include:

- Summary of key findings (adoption lag compression, LLM paradigm shift, predictors)
- Why some research is never adopted (naming, tooling, threat model barriers)
- Implications for researchers and practitioners
- Limitations

]

7 Conclusion

[TODO: Complete after results. Will summarize:

- Key quantitative findings on adoption lag
- Domain-specific patterns
- Acceleration factors
- Contribution of coding framework and dataset

]

A Artifact Source URLs

- **IBM ART:** https://github.com/Trusted-AI/adversarial-robustness-toolbox
- **CleverHans:** https://github.com/cleverhans-lab/cleverhans
- **Foolbox:** https://github.com/bethgelab/foolbox

- **TextAttack:** https://github.com/QData/TextAttack
- **PyRIT:** https://github.com/Azure/PyRIT
- **RobustBench:** https://robustbench.github.io/
- **HarmBench:** https://github.com/centerforaisafety/HarmBench
- **MITRE ATLAS:** https://atlas.mitre.org/techniques
- **OWASP LLM Top 10:** https://owasp.org/www-project-top-10-for-large-language-model-app

References

Anthropic. Detecting and countering malicious uses of Claude: November 2025. https://www.anthropic.com/research/malicious-uses-nov-2025, 2025. First documented large-scale AI-orchestrated cyber campaign; disclosed November 13-14, 2025.

Giovanni Apruzzese, Hyrum S Anderson, Savino Dambra, David Freeman, Fabio Pierazzi, and Kevin Roundy. “real attackers don’t compute gradients”: Bridging the gap between adversarial ML research and practice. In *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 339–364, 2023. doi: 10.1109/SaTML54575.2023.00031.

Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2938–2948, 2020.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022. December 15, 2022.

Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317–331, 2018. doi: 10.1016/j.patcog.2018.07.023.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (SP)*, pages 39–57, 2017. doi: 10.1109/SP.2017.49. arXiv:1608.04644, August 16, 2016.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium*, pages 2633–2650, 2021.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. *arXiv preprint arXiv:2310.08419*, 2024. Introduced PAIR attack.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. RobustBench: A standardized adversarial robustness benchmark. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021. arXiv:2010.09670, October 2020.

Andrea Dal Pozzolo, Olivier Caelen, Reid A Johnson, and Gianluca Bontempi. Calibrating probability with undersampling for unbalanced classification. *IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pages 159–166, 2015. Credit card fraud detection.

European Parliament and Council. Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (artificial intelligence act). Official Journal of the European Union, 2024. Entry into force August 1, 2024; high-risk requirements effective August 2, 2026.

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1625–1634, 2018. doi: 10.1109/CVPR.2018.00175.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015. arXiv:1412.6572, December 20, 2014.

Robert Gorwa, Reuben Binns, and Christian Katzenbach. Algorithmic content moderation: Technical and political challenges in the automation of platform governance. *Big Data & Society*, 7(1), 2020. doi: 10.1177/2053951719897945.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. In *ACM Workshop on Artificial Intelligence and Security (AISec)*, 2023. doi: 10.1145/3605764.3623985. arXiv:2302.12173, February 23, 2023.

Kathrin Grosse, Lukas Bieringer, Tarek R Besold, Battista Biggio, and Katharina Krombholz. Machine learning security in industry: A quantitative survey. *IEEE Transactions on Information Forensics and Security*, 18:1749–1762, 2023. doi: 10.1109/TIFS.2023.3251842. 139 practitioners surveyed; 5% experienced AI-specific attacks, 86% concerned.

Patrick Grother, Mei Ngan, and Kayee Hanaoka. Face recognition vendor test (FRVT) part 2: Identification. Technical Report Interagency Report 8271, NIST, 2019.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. BadNets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama Guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*, 2023. Released December 7, 2023.

Jinyuan Jia, Yupei Liu, and Neil Zhenqiang Gong. BadEncoder: Backdoor attacks to pre-trained encoders in self-supervised learning. *arXiv preprint arXiv:2108.00352*, 2021.

Ram Shankar Siva Kumar, Magnus Nyström, John Lambert, Andrew Marshall, Mario Goertzel, Andi Comissoneru, Matt Swann, and Sharon Xia. Adversarial machine learning – industry perspectives. In *IEEE Security and Privacy Workshops (SPW)*, pages 69–75, 2020. doi: 10.1109/SPW50608.2020.00028. arXiv:2002.05646; 28 organizations surveyed.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018. arXiv:1706.06083, June 19, 2017.

McAfee Advanced Threat Research. Model hacking ADAS to pave safer roads for autonomous vehicles. McAfee Labs Blog, 2020. Published February 19, 2020; 58% success rate with tape modification.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking black-box LLMs with depth-first search. *arXiv preprint arXiv:2312.02119*, 2024. Introduced TAP attack.

Jaron Mink, Harjot Kaur, Juliane Schmid, Sascha Fahl, and Yasemin Acar. “security is not my field, i’m a stats guy”: A qualitative root cause analysis of barriers to adversarial machine learning defenses in industry. In *32nd USENIX Security Symposium*, pages 3763–3780, 2023. 21 semi-structured interviews.

MITRE Corporation. MITRE ATLAS: Adversarial threat landscape for AI systems. https://atlas.mitre.org/, 2024. Launched June 2021; 66 techniques, 33 case studies as of 2024.

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. November 28, 2023; extracted several MB from ChatGPT for $200.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:27730–27744, 2022. arXiv:2203.02155, March 4, 2022.

OWASP Foundation. OWASP top 10 for LLM applications 2025. https://owasp.org/www-project-top-10-for-large-language-model-applications/, 2024. v1.0 August 2023, v2.0 November 18, 2024.

Fábio Perez and Ian Ribeiro. Ignore this title and HackAPrompt: Exposing systemic vulnerabilities of LLMs through a global scale prompt hacking competition. In *NeurIPS 2022 Workshop on Machine Learning Safety*, 2022. arXiv:2211.09527.

Sander Schulhoff, Jeremy Pinto, Anaum Khan, Louis-François Bouchard, Chenglei Si, et al. Ignore this title and HackAPrompt: Exposing systemic vulnerabilities of LLMs through a global scale prompt hacking competition. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024. 137,000+ adversarial interactions, 44 defenses all eventually bypassed.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (SP)*, pages 3–18, 2017. doi: 10.1109/SP.2017.41.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014. ICLR 2024 Test of Time Award.

Tencent Keen Security Lab. Experimental security research of Tesla autopilot. https://keenlab.tencent.com/en/2019/03/29/Tencent-Keen-Security-Lab-Experimental-Security-Research-of-Tesla-Autopilot/, 2019. Published March 29, 2019.

Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction APIs. In *25th USENIX Security Symposium*, pages 601–618, 2016.

Guoming Zhang, Chen Yan, Xiaoyu Ji, Tianchen Zhang, Taimin Zhang, and Wenyuan Xu. DolphinAttack: Inaudible voice commands. In *ACM Conference on Computer and Communications Security (CCS)*, pages 103–117, 2017. doi: 10.1145/3133956.3134052. ACM CCS 2017 Best Paper Award.

Andy Zou, Zifan Wang, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. Spotlight paper; arXiv:2307.15043, July 27, 2023.