ESP-JETA
ESP Journal of Engineering & Technology Advancements
ISSN: 2583-2646/ Volume 6 Issue 4 October-December 2026
Paper Id: JETA-V6I4P109/ Page No: 82-89

Original Article

Adversarial Machine Learning in Production AI Systems: A Systematic Review of Attack Surfaces, Detection Techniques, and Model Resilience

Karthikeyan Konkimalla
Indian Wesleyan University, Merrillville, 4201 S Washington St, Marion, IN 46953

Received Date: 30 October 2026 Revised Date: 05 November 2026 Accepted Date: 11 November 2026

Abstract: *Adversarial machine learning has become a serious threat to production AI systems, not only in the lab, but also once deployed, as the models, training pipelines, APIs, and monitoring layers are exposed. This review examines peer-reviewed journal literature on adversarial attack surfaces, detection techniques, and model resilience, with particular emphasis on production AI environments. Reported research agrees on three primary classes of risk: training-time manipulation, inference-time evasion, and model-facing attacks, in which an attacker exploits the deployed prediction interface. While detection mechanisms utilizing statistical anomalies, input preprocessing, representation analysis and disagreement among classifiers offer protection in carefully limited scenarios, they often fail to be effective against adaptive attackers. Robust training and structure-driven defences tend to offer more resilience than individual detectors, with computational cost, distribution shift, attack transferability, and incomplete threat coverage serving as key limitations. Literature is dominated by benchmark-centred evaluation rather than production evidence over a long period of time. Progress thus requires lifecycle-focused evaluations, adaptive red team processes, operational telemetry, reproducible resilience metrics, and defences based on non-static perturbation assumptions and capabilities of the attacker.*

Keywords: *Adversarial Detection, Adversarial Machine Learning, Attack Surfaces, Model Resilience, Production AI, Robust Learning*

I. INTRODUCTION

Production AI systems have a broader adversarial attack surface than the trained model alone. There are a variety of factors that can change the likelihood or effect of an adversarial failure: data collection methods, labelling, model building, storage of artifacts, interfaces for inference, feedback channels, retraining processes, and monitoring infrastructure. While the research on adversarial-machine-learning focused on crafted test-time inputs, today's security challenge now extends beyond model manipulation to manipulation throughout the entire model life cycle. Biggio and Roli described this evolution as a shift from single pattern recognition attacks to a security field with defined attacker goals, capabilities and knowledge [1]. This perspective is important for production AI, for instance, since deployment turns static models into constantly exposed computational services.

Research in deep learning has led to increased concerns about inference-time manipulation, as classifiers with high accuracy can give drastically different classifications for carefully chosen perturbations. Surveys have revealed significant differences in attacker knowledge, perturbation constraints, optimization strategies, and attack transferability [2]. Yuan et al. also demonstrated that many defensive proposals are hard to compare due to differing protocols for evaluation, differing levels of attack, differing models, and differing assumptions regarding access [3]. This heterogeneity means that it is hard to directly conclude operational resilience from the accuracy of benchmarks under a perturbation family: the security of a benchmark under one family of perturbations does not necessarily imply the security of the same benchmark against an adaptive production adversary.

A production-oriented review should therefore be able to separate algorithmic robustness from system resilience. Algorithmic robustness is about the behaviour of the model in a given set of perturbations; resilience also refers to the difficulty of preventing the attack, detecting the attack, containing the attack, observing the model, recovering from the attack, and retraining the model in a secure way. In fact, it has been shown through taxonomic research that machine-learning attacks may happen during training of the model or during inference, and are capable of compromising confidentiality, integrity, or availability [4]. The production risk is then not only dependent on the sensitivity of the classifiers but also on interactions with the surrounding infrastructure and the behaviour of the model.

A number of issues need to be discussed in more depth. First, available literature usually provides empirical support for claims subsequently extended to other production fields. Secondly, the attacks are usually tested against an attack that

This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/2.0/)

did not specifically target the detector, which may overestimate protection. Third, poisoning and backdoor studies reveal risks that arise prior to deployment, and post-deployment studies often assume that the trained artifact is reliable. Fourth, operational quantities, such as detection delay, false-alert burden, attack prevalence, retraining safety, service degradation and recovery costs are given far less attention than is given to attack success rate or clean accuracy. This review critically examines peer-reviewed journal literature published between 2015 and 2026 related to production-relevant attack surfaces, detection methods, and resilience mechanisms, and to identify methodological conditions that would make the claims of security made by reported protection valid or invalid.

II. LITERATURE REVIEW

The perspectives of journal literature on the adversarial risk come together on the idea of a lifecycle but there are significant differences in the emphasis. Reviews of adversarial examples categorize attacks based on adversary knowledge, optimization techniques used, norm constraints and targeted versus untargeted attack objectives [5]–[9]. This structure can be used to evaluate inference-time evasion, but only partially captures the concept of production exposure since other security states arise from service APIs, feedback loops, data provenance, retraining, and model extraction which cannot be represented by perturbation geometry. Research with a more comprehensive security viewpoint, on the other hand, categorizes attacks based on lifecycle phase of the attack process and security goals, generating more similarities with operational threat modelling [4, 10, 13].

The second literature cluster is related to detection and defence. There are several types of studies, which can be split into detection-oriented studies [3, 11] that include preprocessing, statistical tests, auxiliary detectors, hidden-representation analysis, and prediction-consistency methods. Comparative evidence shows that many detectors, when attacked with distributions similar to their development learning, perform well but degrade dramatically if the attack is optimized using knowledge about the detector. In fact, in defence reviews this limitation is a recurring theme: detection accuracy alone is not enough proof of being resilient to adversarial scenarios [7], [11] and [14]. Defensive evaluation is best done if attacks are adaptive, gradient-aware if applicable and tested across a variety of perturbation budgets and access models.

A third group is about training-time integrity. Adversarial behaviour can be added before the inference, such that a model may have high nominal performance, but exhibit malicious behaviour to specific inputs or corrupted parts of the input distribution [12]. This is particularly impactful for production AI that is built from data from external sources, pre-trained models, or automated retraining pipelines. However, the training-time integrity is not only dependent on the filtering of inferences, but on other factors like proving the provenance, inspecting the dataset and application of a robust training algorithm, as well as testing the trained behaviour.

Table 1: Summary of Key Findings

Ref	Focus	Key Findings
[5]	Adversarial attack and defence taxonomy	Attack effectiveness depends strongly on attacker knowledge, perturbation constraints, and defensive adaptation; static defence comparisons have limited generality.
[6]	Cross-domain adversarial attacks	Images, graphs, and text exhibit domain-specific feasibility constraints, limiting transfer of defence conclusions across modalities.
[7]	Deep-learning attacks and defences	Robust training generally offers broader protection than input-only countermeasures but imposes accuracy and computational trade-offs.
[8]	Object-recognition adversarial examples	Evaluation practices frequently differ in threat model, dataset, architecture, and perturbation budget, complicating comparative robustness claims.
[9]	Opportunities and limitations of adversarial examples	Attack transferability and optimization diversity weaken defences calibrated to single attack families.
[10]	Cybersecurity-domain adversarial ML	Production security requires explicit treatment of feature manipulation constraints, attacker access, and domain-valid modifications.
[11]	Adversarial-example detection	Numerous detectors perform well against non-adaptive attacks; adaptive evaluation exposes weaker reliability for several detector classes.
[12]	Data poisoning and backdoors	Dataset compromise can preserve clean performance while embedding targeted malicious behaviour, making provenance and training-time controls essential.

The following representative journal studies related to production-oriented assessment have been summarized in Table 1. The comparison indicates that the evidence for attack mechanisms is better, and the evidence for end-to-end operational controls is limited. One common methodological challenge is the disconnect between experiments run at the model level and the security level achieved in production.

There are also structural discrepancies between the evidence of offense and defence in the literature. A successful attack construction can result in a failure, while a defence needs to be proven over multiple attacker constructions. A higher evidentiary burden is required for defensive claims as compared to attack claims. This asymmetry is exacerbated in production deployment when the attacker can search over time while the defenders need to have an acceptable latency, false-positive rate, accuracy and availability.

III. METHODOLOGY

There are 4 objects of analysis for methodological approaches: adversarial inputs, training data, learned representations, and deployed interfaces. Input-centred studies define adversarial generation as constrained optimisation and measure attack success, perturbation size, transferability and change in confidence [2, 3, 7]. The poisoning, label manipulation, clean label attacks, and backdoor attacks by modifying training examples and training procedures are analysed in dataset-centred work [12]. Representation-based defences analyse the activations, the consistency of predictions, the reconstruction behaviour, or auxiliary statistics [11]. Black-box probing, feasibility in feature space, API access and operational constraints are taken into account in interface-centred security work [4], [10], [13].

These methods vary in the strength of the evidence. Explicit threat assumptions enable optimization-based attack experiments, which offer reproducible testing; however, stress tests based on such narrow norms do not necessarily represent semantic or physically realizable manipulation. When the distribution of the attacks is known in advance of designing the detector, conventional measures, such as true-positive rate, area under receiver-operating-characteristic curve, and false-positive rate, are derived from the results of the detector benchmarking. Strong-training methods directly modify decision behaviour, and as a result do not require the use of an independent detector, but the level of resilience is still constrained by the model of the perturbation to be learned during training [7, 14]. In a sense, training-data defences cover a previously unmentioned phase, but they have to be able to tell the difference between adversarial contamination and rare but legitimate observations.

To illustrate the methodological families of these different approaches, Table 2 presents the principal methodological families reported in the literature. The overarching principle is that more secure claims emerge when a method is tied to a clearly specified threat model to adaptively evaluate the method, instead of having a nominal high benchmark performance.

Table 2: Method Comparison

Ref	Method	Strengths	Limitations
[1]	Security-oriented threat modelling	Separates attacker goals, knowledge, and capabilities; supports lifecycle reasoning.	Provides a framework rather than a deployment-specific effectiveness estimate.
[3]	Attack/defence benchmark comparison	Enables systematic comparison of perturbation and defence families.	Reported results remain sensitive to attack implementation and parameterization.
[7]	Robust optimization and defensive training	Changes model behaviour directly and can improve worst-case performance.	Computational burden and clean-accuracy trade-offs can be substantial.
[11]	Detector benchmarking	Measures discriminative ability across multiple detector families.	Detector performance can overstate security without adaptive attack evaluation.
[12]	Poisoning/backdoor analysis	Captures pre-deployment integrity failures absent from evasion-only frameworks.	Trigger assumptions and contamination models vary markedly across studies.
[14]	Defense-centered comparative review	Emphasizes adaptive attacks, evaluation pitfalls, and robustness criteria.	Published comparisons remain heterogeneous across datasets and architectures.
[15]	Cross-attack defence taxonomy	Links evasion, poisoning, and defensive mechanisms within a common classification.	Broad coverage limits depth for production observability and recovery mechanisms.

The methodological families are arranged in a production-AI lifecycle in Fig. 1. It is important because when an adversarial scenario presents itself, it can occur before training, during inference, and in post-deployment interaction, all of which require a different model component to be adversarial resilient.

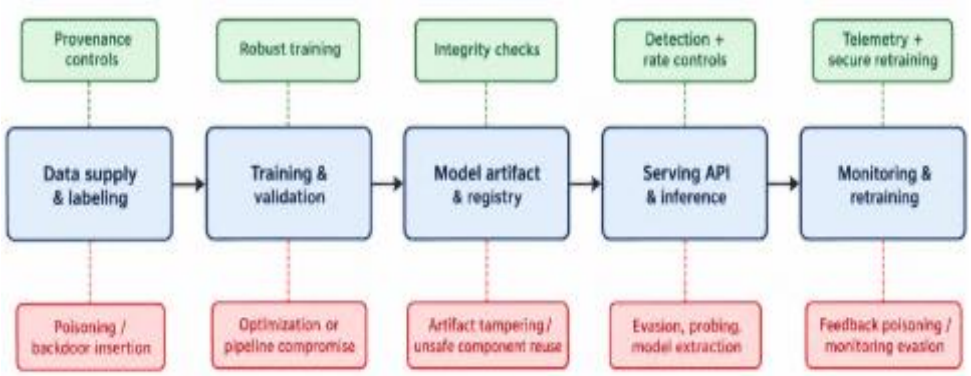


Figure 1. Production-AI Adversarial Attack Surface and Resilience Control Points.

IV. DISCUSSION

The most consistent cross-study result is that neither attack success nor defence effectiveness is an inherent property of a model. Evasion results are influenced by the knowledge of the attacker, the perturbation budget, the optimization procedure, query access and semantic constraints [2] [3] [5]–[9]. Therefore, single-number robustness claims should not be interpreted as evidence of production resilience without a clearly specified threat model. A bounded white-box perturbation-learned system with high adversarial accuracy can still be susceptible to black-box transfer, restricted semantic changes, physical manipulation, or attacks on the upstream features. On the other hand, an attack that has a high benchmark success rate might be operationally insignificant if it forces features to be changed, which doesn't meet the domain constraints. In cybersecurity applications, Rosenberg et al. demonstrate that in order to generate features while maintaining the malicious function, the changes made to the model must not be invalid [10].

Table 3: Results Comparison

Ref	System / Context	Metric / Basis of Comparison	Outcome
[2]	Deep image classifiers	Attack success, perturbation constraint, transferability	Small, constrained perturbations can produce high attack success across multiple architectures; transfer limits reliance on obscurity.
[3]	Deep-learning classifiers	Adversarial accuracy and attack/defence comparison	Defence rankings vary with attack type and strength; no single mechanism dominates across threat models.
[7]	Deep neural networks	Robust versus clean predictive performance	Robust-training families improve resistance inside modelled perturbation regions but incur computational and accuracy trade-offs.
[10]	Cybersecurity ML	Domain-valid feature manipulation	Realistic attack feasibility depends on preserving malicious functionality, reducing the value of unconstrained feature-space attacks.
[11]	DNN adversarial detectors	Detection performance under multiple attack conditions	Detector effectiveness is highly attack-dependent and can decline under stronger or adaptive evaluation.
[12]	Training datasets and backdoored models	Clean performance versus triggered behaviour	Backdoored models can preserve nominal accuracy while producing attacker-selected responses on trigger-bearing inputs.
[14]	DNN defence evaluation	Adaptive attack success and robustness criteria	Several apparent defences provide weaker protection after optimization artifacts and attack adaptation are controlled.
[15]	Cross-domain adversarial ML	Attack class versus defence class	Defence coverage is fragmented; mechanisms addressing one attack surface frequently leave other lifecycle stages exposed.

Table 3 shows a comparison of outcomes based upon the most production-interpretation-relevant basis. The table is designed to highlight the differences between empirical discrimination, adversarial accuracy, transfer behaviour and lifecycle coverage since these are not synonymous measures of resilience.

A second consistent finding is related to adaptive evaluation. Detection mechanisms may seem very successful if adversarial examples are created without knowledge of the detector, particularly if attacks create detectable features in the input statistics or internal representations [11]. If an adversarial sample successfully embeds the detector into the

optimization objective, or approximates the behaviour of the detector, separation between clean and adversarial samples can become thin. Miller et al. further emphasizes that a security assessment of defence should take into consideration adaptation by attacks and obfuscation of gradients, rather than treating failure under non-adaptive attacks as sufficient evidence of defensive robustness [14]. The difference is important for production AI, since the deployed defences are persistent targets: repeated queries can provide enough behavioural information to support adaptation without having access to the internal models.

Compared to other post hoc defences, such as recognizing an adversarial signature, robust training is stronger because it changes the learned function, not just try to identify a known adversarial signature [7, 14]. However, the literature does not support the interpretation of strong training as a "production defence." Robustness can be optimized within a specified perturbation set, but such optimization does not guarantee protection against attacks outside that set. The cost of training also increases since adversarial examples need to be either created or approximated during optimization. The clean accuracy degradation can be tolerated in safety-critical applications but be important in high volume services. Whereas the question in the production is not whether strong training works or not, but which failure region is shrunk at what operational cost, and which attacker capability is reduced.

The training-time compromise adds a new type of resilience issue. Data poisoning and backdoor attacks can introduce targeted malicious behaviour while leaving aggregate model performance largely unaffected, making such compromises difficult to detect through conventional validation. Conventional validation can then fail to validate the targeted behaviour, if the validation data do not contain the trigger or poisoned subpopulation. This property undermines the assurance strategies that rely solely on the accuracy of the held-out. Monitors may be needed for production systems that depend upon external datasets, pretrained components, continuous learning, or user feedback, as inference-time monitoring starts after some attacks have been successful.

The distribution of topics across the reviewed journal set was captured quantitatively using transparent qualitative coding as shown in Figure 2. The imbalance is significant as evasion is being discussed heavily during inference time while production monitoring and operational recovery are not. The added significance is that there is a gap between evidence of robustness at the benchmark level and the actual level of resilience in deployment.

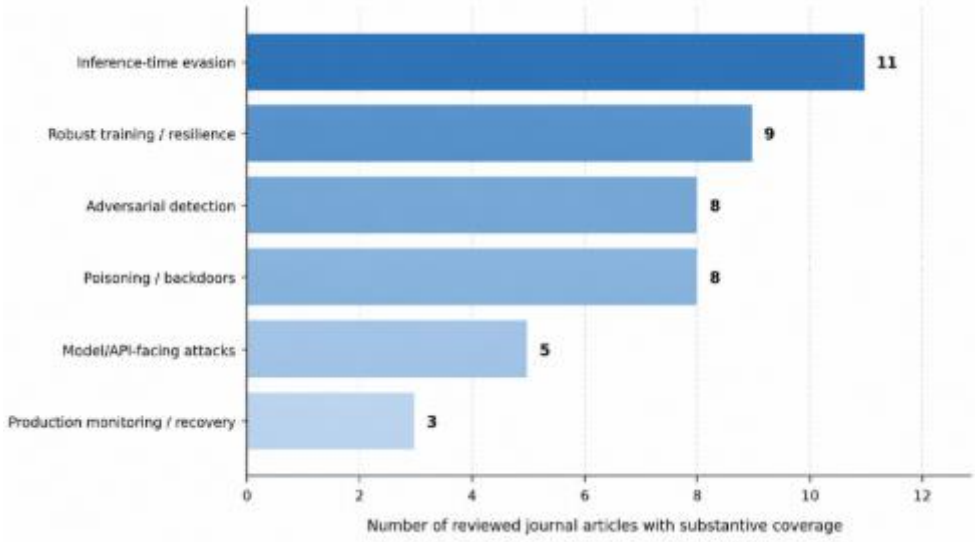


Figure 2: Distribution of Production-Relevant Adversarial-ML Themes Across The Reviewed Journal Literature.

Scientific quality and the magnitude of effect are not measured in the coded distribution. Rather, it shows where the journal is focused. Evidence in the production monitoring & recovery category is the smallest, though important after deployment. This mismatch contributes to the lack of guidance provided by the adversarial robustness benchmarks for incident detection, rollback criteria, retraining controls, and service continuity.

Another cross study trend relates to transferability. It spreads attacks from surrogate models to target systems which are not white-boxed [2] [3] [9]. This is a property that breaks the security argument of the architecture secrecy and limited access to the model property in production. Attacker cost can be raised by implementing rate limiting and query controls, but these controls will not stop attacks built offline against the substitutes. Transfer poses another challenge in detector evaluation because it is possible for perturbations to pass through regular inference channels into a detector without actually interacting with it when building the attack.

Resilience ought hence to be described as a multi-layer characteristic instead of a defensive status. Preventive mechanisms include data provenance, access control, robust training, and model-artifact integrity. Detection mechanisms include runtime anomaly detection, consistency checks, confidence analysis, distribution monitoring, and security telemetry. Containment involves restriction, abstention, model fallback, quarantine or temporary isolation of suspect inputs. Validation of retraining, rollback of artifacts, trigger remediation and post incident verification were all part of recovery. The reviewed literature predominantly addresses attack mechanisms and preventive defences, whereas containment and recovery remain comparatively underexplored.

Figure 3 combines defence families based on the defence family's lifecycle position and evidentiary assurance. The visual is important because post hoc detectors, robust learning, provenance controls and operational telemetry address distinct security problems, and using the same technique for each problem ignores the existence of any attack surfaces.

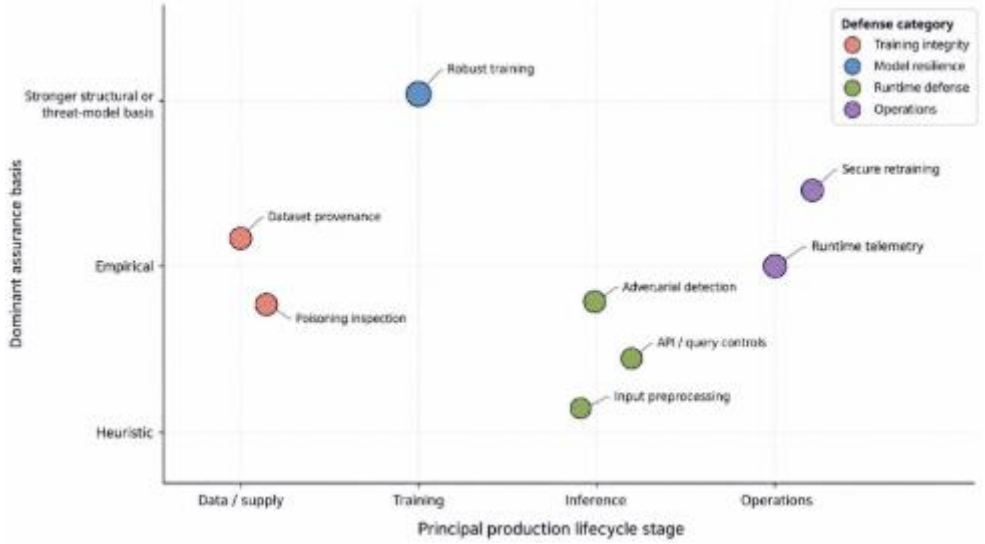


Figure 3: Relational Map of Adversarial-Defence Families by Lifecycle Stage and Assurance Basis.

A significant difference between empirical assurance and structural assurance is shown in the figure 3. Runtime detectors tend to be tested empirically using specific sets of attacks, whereas provenance controls and strong optimizations obtain some of their usefulness from limiting the opportunities of the attackers, or explicitly optimizing with respect to a threat model. Neither approach independently provides comprehensive protection against the full range of production attack scenarios. Structural safeguards still might represent an incomplete assumption, while empirical detectors might be able to detect events that are not represented by formalized perturbation models.

There are three general implications of this. Evaluation of production should include attack-conditioned performance and ordinary operating performance together with the first aspect. Security relevant are false positives, abstentions, latency, computational load, and clean accuracy, as they can lead to the disablement or circumvention of the controls if the cost of the security controls is too high. Second, it should be noted that assessing resilience should be distinguished from known-attack benchmarking and adaptive red team testing. The former is used to determine if it's possible to repeat the experiment; the latter is used to see if the control is useful after the attacker adapts. Third, there is a need to evaluate security feedback loops in continuous-learning systems. Earlier flagging of suspicious inputs should not be interpreted as them automatically being retraining examples, as the attacker could use the pipeline for monitoring in order to set up a secondary poisoning channel.

There is also no standardized evidence found in the literature in terms of cross-layer composition. A combination of strong training, anomaly detector, provenance checks and query throttling might be more resilient than any of their components considered individually, but no independent metrics about these components can determine the failure probability of the combination. Defensive layers can also have a negative interaction with one another. Preprocessing can bias distribution monitoring signals, abstention can provide a side channel, and an aggressive automated retraining could turn an evasion campaign into data contamination. To evaluate the production, therefore, the experiments must be integrated, allowing the AI service to be used as a security system and not as a classifier to which filters are applied.

V. FUTURE DIRECTIONS

Threat models that indicate the attacker's access, modification restrictions, query budget, attacker's adaptation capabilities, persistence, and targeted production asset are important ones in the future that should be considered. Robustness metrics should accompany such models to be able to interpret adversarial accuracy, detector performance or

poisoning resistance with respect to a specific security boundary [1, 14]. Additionally, the following lifecycle transitions should be included in the benchmark suites since they can't be captured by static inference benchmarks: data ingestion, model update, rollback, monitoring.

There is a need for greater standardization for adaptive evaluation. The false positive rate of detection studies should be reported at realistic base rates, attack success after adaptation, computational latency, and performance in benign distribution shift [11]. Studies of robust training should include a report on a family of training perturbations that is not the family of interest for the protection, and they should provide a separation of reduced attack success from attack-detection capability [7]. Realistic data-supply chains, pretrained components, federated or continuously updated models and imperfect provenance should be included in poisoning and backdoor research [12].

Resilience following successful compromise is a second priority. Research should consider containment policies, safe abstention, fallback models, artifact rollback and validated retraining as possible options rather than attack prevention as the only achievable end point. A combination of security engineering, reliability engineering, machine learning, and human factors should be used to study the ability of alerts to elicit timely and correct operational responses. Incident-oriented datasets may be utilized to evaluate detection delay, analyst burden, recurrence, and recovery quality.

Lastly, comparative analysis should be conducted on composed defences in the face of common adversaries. A production-relevant benchmark would evaluate the data controls, strong learning, runtime detection, service controls and recovery procedures as an integrated system. This assessment would shift the focus from isolated attack-success rates to end-to-end system resilience against adaptive adversaries.

VI. CONCLUSION

The important point to understand about adversarial machine learning in production AI systems is that it is not a property of the classifiers' sensitivity, but rather a lifecycle security problem. The analysis of the reviewed literature indicates a mature understanding of inference-time evasion, increasing awareness of poisoning and backdoor threats, and great experimentation with detection and robust-learning mechanisms. However, evidence is not consistent throughout the production attack surfaces. Empirical focus on runtime monitoring, containment, recovery, safe retraining, and defence against multiple layers is significantly less than the focus on adversarial example generation and benchmark robustness.

None of the isolated defence families can be used to guarantee production assurance. Input detectors can break down after attackers are adapted, robust training is limited to modelled perturbation regions, and nominal validation may not catch targeted training-time compromise. Explicit threat modelling, training data integrity, model-level robustness, adaptive runtime detection, service controls, and controlled recovery procedures are therefore dependent on each other and must be implemented as complementary layers to strengthen resilience.

The gap between benchmark robustness and evidence of operational security is the principal scholarly gap. Future advancement will rely on evaluation protocols that evolve to provide a system a variety of adaptive attackers throughout its lifecycle and measure the effectiveness of security, clean performance, latency, false-positive burden and recovery behaviour. Such a transition would enable more robust claims of the reliability of production AI systems under conditions of ongoing adversarial interaction outside of the laboratory.

VII. REFERENCES

[1] Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition, 84*, 317–331.

[2] Akhtar, N., & Mian, A. (2018). Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access, 6*, 14410–14430.

[3] Yuan, X., He, P., Zhu, Q., & Li, X. (2019). Adversarial examples: Attacks and defences for deep learning. *IEEE Transactions on Neural Networks and Learning Systems, 30*(9), 2805–2824.

[4] Pitropakis, N., Panaousis, E., Giannetsos, T., Anastasiadis, E., & Loukas, G. (2019). A taxonomy and survey of attacks against machine learning. *Computer Science Review, 34*, 100199.

[5] Qiu, S., Liu, Q., Zhou, S., & Wu, C. (2019). Review of artificial intelligence adversarial attack and defence technologies. *Applied Sciences, 9*(5), 909.

[6] Xu, H., Ma, Y., Liu, H.-C., Deb, D., Liu, H., Tang, J.-L., & Jain, A. K. (2020). Adversarial attacks and defences in images, graphs and text: A review. *International Journal of Automation and Computing, 17*, 151–178.

[7] Ren, K., Zheng, T., Qin, Z., & Liu, X. (2020). Adversarial attacks and defences in deep learning. *Engineering, 6*(3), 346–360.

[8] Serban, A., Poll, E., & Visser, J. (2020). Adversarial examples on object recognition: A comprehensive survey. *ACM Computing Surveys, 53*(3), Article 66.

[9] Zhang, J., & Li, C. (2020). Adversarial examples: Opportunities and challenges. *IEEE Transactions on Neural Networks and Learning Systems, 31*(7), 2578–2593.

[10] Rosenberg, I., Shabtai, A., Elovici, Y., & Rokach, L. (2021). Adversarial machine learning attacks and defence methods in the cyber security domain. *ACM Computing Surveys, 54*(5), Article 108.

[11] Aldahdooh, A., Hamidouche, W., Fezza, S. A., & Deforges, O. (2022). Adversarial example detection for DNN models: A review and experimental comparison. *Artificial Intelligence Review, 55*, 4403–4462.

[12] [Goldblum, M., Tsipras, D., Xie, C., Chen, X., Schwarzschild, A., Song, D., Mądry, A., Li, B., & Goldstein, T. (2023). Dataset security for machine learning: Data poisoning, backdoor attacks, and defences. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45*(2), 1563–1580.

[13] Liu, Q., Li, P., Zhao, W., Cai, W., Yu, S., & Leung, V. C. M. (2018). A survey on security threats and defensive techniques of machine learning: A data driven view. *IEEE Access, 6*, 12103–12117.

[14] Miller, D. J., Xiang, Z., & Kesidis, G. (2020). Adversarial learning targeting deep neural network classification: A comprehensive review of defences against attacks. *Proceedings of the IEEE, 108*(3), 402–433.

[15] Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., & Mukhopadhyay, D. (2021). A survey on adversarial attacks and defences. *CAAI Transactions on Intelligence Technology, 6*(1), 25–45.