Exploring Perturbation Patterns and Impact in Adversarial Machine Learning: A Systematic Literature Review

Alexandra Sheykina
Department of Computer Science
SeSa Lab - University of Salerno
Fisciano, Italy
asheykina@unisa.it

Fabio Palomba
Department of Computer Science
SeSa Lab - University of Salerno
Fisciano, Italy
fpalomba@unisa.it

Andrea De Lucia
Department of Computer Science
SeSa Lab - University of Salerno
Fisciano, Italy
adelucia@unisa.it

***Abstract*—Adversarial attacks have gained growing attention due to their ability to mislead machine learning models by introducing carefully crafted perturbations. These attacks span a wide variety of domains, from image recognition to graph-based and NLP models. In this context, understanding the nature of such perturbations is crucial for both detecting attacks and designing effective defenses. Despite the abundance of research on adversarial machine learning, little is known about how perturbation types vary based on the category of the targeted feature, or how the amount of perturbation influences the attack's impact. To this aim, we conducted a systematic study to (i) identify and classify the perturbation strategies used in adversarial attacks and (ii) analyze the relationship between the strength of perturbation, the perturbation type, and the impact on model behavior. Our findings show that while many attacks apply minimal but targeted changes, the perturbation type plays a major role in determining the success of the attack. Furthermore, attacks with similar perturbation magnitudes may have vastly different impacts depending on their semantic focus. These insights can support the prioritization of defense mechanisms by focusing on high-impact perturbations, and lay the groundwork for improved adversarial detection systems based on perturbation-level analysis.**

***Index Terms*—Adversarial Machine Learning; Software Security; Systematic Literature Reviews.**

I. Introduction

Over the past 20 years, Machine Learning (ML) has become integral to various devices, services, and systems, revolutionizing many sectors. Examples include virtual assistants like Apple's Siri, Amazon Alexa, and traffic prediction using Google Maps. Additionally, innovations like a smart camera using TensorFlow Light and DeepMind's AlphaFold, which solved the protein folding problem, showcase ML's potential. ML has also advanced disease prevention, agriculture, and cancer treatment through molecular and genetic profiling.

However, as ML systems become more pervasive, they also introduce specific vulnerabilities, making them susceptible to Adversarial Attacks (AAs). These attacks target critical infrastructures such as autonomous vehicles and medical devices, posing significant threats to human safety and public security. Furthermore, the financial and reputational risks, along with intellectual property theft and legal issues, make it increasingly essential to secure ML models.

Given the increasing reliance on ML systems, it is crucial to assess the vulnerability of these models to adversarial perturbations. This study aims to investigate how different types of perturbations (e.g., pixel shifts, word substitutions, or more complex transformations) impact model behavior and security. By exploring these perturbations, we seek to identify vulnerabilities in ML models and gain insights that can guide the development of more effective defense strategies.

We reviewed 162 relevant studies, mostly recent. The *purpose* is to understand how different types of perturbations and their characteristics affect the vulnerability of ML models. We adopt the *perspective* of both researchers, interested in modeling and detecting adversarial behaviors, and practitioners, focused on prioritizing defense strategies and ensuring the reliability of ML models. The *goal* is to examine the relationship between perturbation magnitude, type, and their impact on model security. The study highlights patterns across attack methods, discusses implications for detection and defense prioritization, and identifies open challenges in systematically integrating these insights into robustness evaluation tools for ML systems.

Structure of the paper. Section II introduces the background of AML and summarizes the related work. Section III describes the research design and research questions of our systematic literature review, while the results are presented in Section IV and the threats to validity in Section V. Finally, Section VI concludes the paper and reports future work.

Online appendix. All the data collected and the complete analysis conducted in this study are available in the replication package [1].

II. Related Work

Research in the AML has evolved significantly, progressing through four stages, from basic attack and defense methods to more sophisticated techniques and applications. In the initial stage, research focused on attacks and defenses in computer vision and cybersecurity, using classical ML models such as Random Forest, Support Vector Machine (SVM), and Linear Regression. A pioneering analysis by Barreno et al. [2] examined ML security, categorizing various attacks and defenses. Liu et al. [3] further analyzed security threats during both the training and testing phases, classifying defensive techniques into four groups: security evaluation mechanisms, countermeasures in the training phase, those in the testing or inference phase, data security, and privacy.

With the advent of DL, research shifted towards adversarial methods specific to deep models. This period witnessed the development of numerous attack and defense algorithms for deep learning. Akhtar et al. [4, 5] conducted a comprehensive survey on AAs in DL, proposing defenses and evaluating real-world scenarios. Their study spanned tasks like vision recognition, malware detection, and speech recognition, highlighting the transferability of attacks between neural networks. They also analyzed the attacks on different neural network models, such as Convolutional Neural Network (CNN)-based classification, Recurrent Neural Networks (RNN), and Deep Reinforcement (DR) learning. Biggio et al. [6] explored the security properties of ML algorithms in the computer vision and cybersecurity domain, focusing on security evaluation.

In the third phase, AML research began to expand into various areas, tasks, and ML and DL algorithms. This includes applying adversarial techniques not only to images but also to text [7], audio, video, graphs [8, 9], and time series. In this period, numerous attack taxonomies have been proposed [10, 11, 12, 13, 14] against ML [15] and DL algorithms for various applications: spam filtering, intrusion detection, visual recognition [16], malware detection [17] and corresponding defenses [9, 18, 13, 19, 20, 21, 22, 23]. Empirical studies have been conducted for different tasks such as object recognition [14], speech recognition, text classification, image classification, malware detection [8], face recognition [13], text classification, sentiment analysis [22], PE malware detection [23]. Furthermore, different types of attacks were analyzed, for example, poisoning [20, 21], backdoors [24], evasion [25], inference, and inversion attacks [26]. Various defense techniques have been proposed. Chen et al. in [27] investigated adversary attacks and their defenses in DR learning under artificial intelligence security. Qiu et al. [28] analyzed attack methods in training and testing in various domains, while some authors reviewed adversarial techniques to improve model robustness [29, 23].

Our contribution. While prior research in AML has extensively explored attack strategies, defense mechanisms, and even proposed taxonomies of adversarial examples, fewer studies have focused on systematically characterizing how different types of perturbations influence model vulnerability and security outcomes. Our work aims to fill this gap by offering a comprehensive analysis of the relationship between perturbation patterns and their impact on ML model robustness across multiple domains and tasks.

III. Research Methodology

Our paper investigates the current landscape of Adversarial Machine Learning (AML) approaches to characterize the types of perturbations applied in attacks and their impact on model robustness. We aim to answer the following overarching research question:

What characteristics of adversarial attacks influence model behavior, and how can this information be used to support detection and defense mechanisms?

which has been detailed into two sub-questions:

- **RQ**$_1$: *What are the main perturbation patterns that emerge across adversarial attacks?*
- **RQ**$_2$: *How do the type and perturbation strength influence the security impact on ML models?*

To formulate the database search string, we identified keywords from our research questions. Each provisional search string was then validated against a list of relevant primary studies, as suggested in the guidelines [30]. The process ended when all known studies were included, the number of retrieved documents was manageable, and all relevant keywords were present. We relied on the three stages defined by Kitchenham et al. [31]: (i) elaborate the search string, (ii) apply the string on chosen search engines, (iii) filter out and extract the studies based on inclusion and exclusion criteria.

The search string is based on the GQM terms to define the research goal by focusing on purpose, issue, object, and viewpoint [32].

Purpose: systematically categorize the types of perturbations
Issue: perturbations and defense strategies
Object (process): adversarial attacks, detection, and defenses
Viewpoint: from the researchers' perspective
The search query derived from the **RQ**s is the following:

(machine learning ∨ neural network ∨ deep learning) ∧ (adversarial sample ∨ adversarial perturbation* ∨ adversarial example*) ∧ (misclassif* ∨ robustness ∨ vulnerability) ∧ (attack ∨ defense) ∧ (algorithm ∨ technique)*

We applied the search query to different databases *Scopus*[1], *IEEEXplore*[2], and *ACM Digital library*[3], to search for articles related to our work. We executed the query string on three scientific databases, applying the filter parameters where possible. The search query produced a total of 2,994 papers for Scopus, 1,093 papers for IEEE, and 1,227 papers for ACM, obtaining 5,314 studies. We loaded all the collected articles into a local database. The results were screened against inclusion and exclusion criteria. In the study selection phase, we applied both automatic and manual modes for selecting relevant articles using a set of selection criteria to refine the articles resulting from the database search phase. Table I illustrates the complete exclusion and inclusion criteria list. The first author applied

[1]Link to *Scopus*: https://scholar.google.com/
[2]Link to *IEEEXplore*: https://ieeexplore.ieee.org/
[3]Link to *ACM Digital library*: https://dl.acm.org

the exclusion criteria MEC_1 - MEC_5, with reviews by co-authors, resulting in 2,901 articles from three databases. Next, we assigned Scimago rankings to journals and CORE rankings to conferences. To define the rank of the conferences in MIC_3, we referred to *CORE*[4], and for quartiles, we referred to *SCIMAGO*[5], reducing the selection to 1,821 articles. We then applyed the manual exclusion criteria by analyzing titles, abstracts, keywords, and metadata for MEC_6 and resulting in 1,525 articles. With many articles remaining after the first phase, we used a scoring system based on citations, venue importance, and topic relevance. A score ranging between 0 and 1 was calculated for MIC_4, based on the number of citations **A** (from 0 to 3), importance of the venue **B** (1 or 2 score), and relevance to the main topic **C** (from 0 to 5 score), and the articles with a score of at least 0.6 were included, resulting in 189 primary studies. After selecting our primary studies, we conducted the snowballing phase using Google Scholar, applying both forward and backward techniques in a single iteration. We considered the *Google Scholar*[6] database to retrieve relevant documents in the snowballing stage [33]. This identified 45 additional works, which, after applying inclusion and exclusion criteria, resulted in 40 additional studies, totaling 229.

TABLE I
Exclusion and inclusion criteria. Automatic Inclusion Criteria (AIC), Automatic Exclusion Criteria (AEC), Manual Inclusion Criteria (MIC), and Manual Exclusion Criteria (MEC)

Papers written in English	AIC_1
Subject areas are computer science and engineering	AIC_2
We excluded short papers and considered only full research papers	MEC_1
Papers whose full-text read was not available	MEC_2
Conference papers later extended to journal	MEC_3
Unpublished but preprint available in open access repositories	MEC_4
No secondary studies	MEC_5
Formal Method, Testing, Federated learning	MEC_6
Remove duplicated papers	MEC_7
Empirical studies	MIC_1
Published in peer-reviewed journals or conference proceedings	MIC_2
Published in conferences with A and A ranks, and journals with rank equal to Q1*	MIC_3
Apply a score based on the number of citations, importance of venue, and relevance to the main topic	MIC_4

A quality assessment was conducted on a set of 229 primary studies. We answered two research questions to assess the quality of the primary studies:

Q_1: *Are the attack or defense techniques clearly defined?*

Q_2: *Are evaluation metrics applied to measure robustness of model?*

We created a checklist for quality assessments, with questions answered as *Yes*, *Partially*, *No*. Each label was assigned a numerical value: *1* for *Yes*, *0.5* for *Partially*, and *0* for *No*. The overall quality score was the average of these values for the two questions. Articles scoring 0.75 or higher were accepted. At the end of the process, we obtained 162 articles.

Finally, to answer our research questions, we extracted the main characteristics of ASRs and defense methods by analyzing the documentation and experimental setups reported in the studies. During the data extraction phase, we carefully read and examined the selected articles. We used a tabular data extraction form to systematically record the relevant information retrieved from each primary study, enabling us to answer our research questions in a structured way. Specifically, for each study, we collected data on the application domain, the addressed task (e.g., image classification, object detection), the input type (e.g., image, text, graph), the dataset used and its nature (in-vitro, synthetic, in-field), the features involved, the model architecture, the type of perturbation pattern applied, the percentage of perturbation, and the impact on the model's performance (e.g., accuracy drop, misclassification rate increase). An extraction form was completed for each article to capture all the collected information.

In terms of reporting, we followed the *ACM/SIGSOFT Empirical Standards*[7] and, in particular, the *"General Standard"* and *"Systematic Reviews"* guidelines.

IV. Results and Discussion

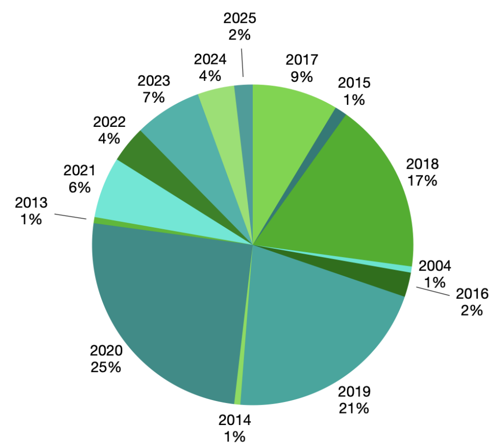

Fig. 1. Study distribution by year.

Figures 1 and 2 illustrate the distribution of the selected 162 studies by year and by domain. The former distribution shows a steady increase in contributions over time, highlighting the growing interest in AML and perturbation techniques. The latter distribution indicates that the selected studies span a wide range of domains, including computer vision, natural language processing, and cybersecurity, reflecting the interdisciplinary relevance of AAs and defenses. In the following sections, we present the key findings and insights derived from our analysis. For sake of page limitations, the full list of 162 primary studies is not included in the paper, but is available in our online appendix [1].

A. *RQ_1: What are the main perturbation patterns that emerge across adversarial attacks?*

To systematically examine how adversarial examples affect model robustness, we identify and describe the main

[4]The *ICORE Conference Portal*: https://portal.core.edu.au/conf-ranks/
[5]The *Scimago Journal Ranking*: https://www.scimagojr.com/
[6]Link to *Google Scholar*: https://scholar.google.com
[7]The *ACM/SIGSOFT Empirical Standards*: https://github.com/acmsigsoft/EmpiricalStandards

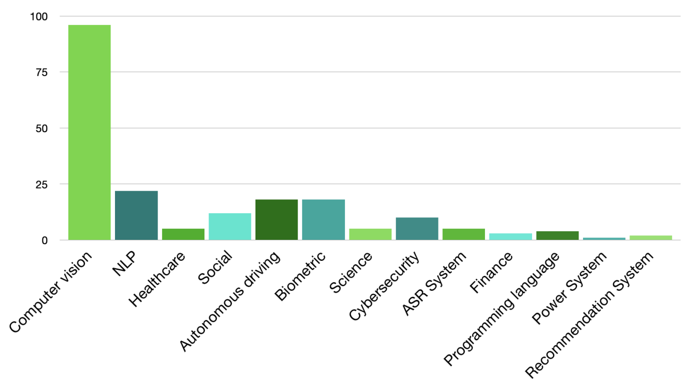

Fig. 2. Study distribution by domain.

perturbation patterns used in AAs across different domains. These patterns represent the characteristic ways in which adversarial modifications are introduced into inputs, models, or their environments. By analyzing 162 studies, we classify perturbations into 14 principal patterns, based on the nature, granularity, and modality of the manipulated data. Specifically, the 14 perturbation patterns are organized into two overarching categories: *input data perturbations*, where the attack modifies the inputs provided to the model, and *model perturbations*, where the adversary alters model parameters, features, or internal structures.

Pixel-level perturbation patterns are among the most common and extensively studied forms of AAs [S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15]. These perturbations typically involve pixel-level and gradient-based manipulations, most often constrained by the L_∞ norm with very small intensities (e.g., below 1.0/255, typically 0.2/255 to 5/255). Some attacks extend to color and texture manipulations [S16, S17, S18, S19, S20, S21, S22], classified as unrestricted semantic manipulations [S23], spanning different norms L_0, L_1, L_2, and L_∞ and ranging from subtle distortions to stronger, less realistic patterns (e.g., textured or sparse color clusters). A significant focus is on minimal perturbations that still compromise complex models, typically with $\epsilon = 0.03$ (corresponding to 8/255) [S24], to remain below human perception thresholds [S25]. Larger perturbations, such as $\epsilon = 128/255$ (approximately 50% of the pixel value range), aim at style-based attacks, altering a substantial part of the pixel range. Several studies introduce *structured perturbations*, including grid-based occlusions and manipulations of the most significant bits (MSB), particularly the first three bits, which strongly affect model performance. SquareAttack applies contiguous square-shaped perturbations (up to 8/255), modifying 5–10% of an image area. Beyond digital attacks, *physical perturbations* (e.g., adjustments to contrast, brightness, Gaussian blur, and noise) have been explored to maintain adversarial effectiveness under real-world conditions. Advanced techniques leverage *gradient priors* to focus perturbations on sensitive image regions, improving attack efficiency while minimizing visible artifacts [S26, S27, S28, S29, S30, S31, S32, S33, S34, S35, S36, S37, S38, S39, S40, S41, S42, S43, S44, S45, S46]. This diversity in perturbation strategies underscores both the complexity and the pervasiveness of adversarial threats, whose characteristics vary significantly depending on the application context and the specific attack techniques employed [S16, S47, S48, S49, S50, S51, S52, S53, S54, S55, S56, S57, S58, S59, S60, S61, S62, S63, S64].

Packet- and byte-level perturbation refers to attacks that work by modifying specific bytes in a file or communication, without altering the semantic content. This type of attack is often used in contexts where data is represented as sequences of bytes, such as in PDF files, network packets, or data streams transmitted between systems. The idea is to add or modify a small number of bytes in such a way that the attack evades detection and does not affect the visual behavior or content of the data (e.g., a PDF document or an executable file) [S65, S66, S65, S67, S68]. Adversarial perturbations typically focus on imperceptible changes involving features like the number of bytes sent and received, the size and count of packets, the number of connections, the average connections per IP, and statistics related to the ratio of bytes exchanged. In the case of documents, they may involve the insertion of bytes without altering the semantic or visual content.

Geometric perturbation pattern acts by directly modifying the spatial arrangement of the image or the represented object. It includes operations such as rotation, translation, scaling, perspective distortion, cropping, and padding [S69, S70, S71, S72, S73]. Some of these transformations are applied digitally, some of these transformations are applied digitally, others simulate realistic physical scenarios, including variables such as distances (from 1m to 25m), camera angles ($-60°$ to $60°$) [S74], indoor/outdoor settings, and lighting variations. Finally, some experiments employ discrete transformations such as moving 0, 1, or 2 pixels in cardinal directions and specific angular rotations (e.g., $-30°$, $0°$, $30°$ for MNIST). More sophisticated attacks adopt instance-wise strategies and act on combined transformations to maximize effectiveness while maintaining high stealthiness [S75, S76, S76].

Bytecode-level perturbation pattern directly intervenes on an app's representations (e.g., Android dex files), modifying features such as permissions, APIs, intents, and components to confuse classifiers. The alterations can affect both the structure visible in the manifest (e.g., unused permissions) and the semantic behavior in the bytecode (e.g., never-executed API calls) [S77, S78, S79, S80].

Word-level perturbation pattern acts on single words or short sequences, preserving the semantic and/or syntactic meaning of the text. It includes substitutions based on synonyms and sememes, minimal units of conceptual meaning, to generate linguistically correct but deceptive variants [S81, S82, S83, S84, S85, S86, S67, S87, S88, S89, S90]. Other techniques involve morphological perturbations (e.g., tense, number), adding or removing nonessential words, and manipulating tokens. In source code tasks such as code comment generation, clone detection, classification, code generation, and summarization, semantic-preserving perturbations similarly target identifiers or snippets, typically performing 2–3 substitutions per code to rename variables, functions, or classes while maintaining syntactic and semantic correctness. These renamings may involve long or semantically distant names to complicate tokenization (e.g., CodeBERT) and reduce performance, with notable effectiveness when applied within control structures (e.g., for, if). Style and structural transformations combine identifier changes with operations like inserting log statements, swapping while and for loops, reordering binary conditions, adding try-catch blocks, inserting dead code, or propagating boolean changes. Additional semantically-preserving techniques include dependency-free statement permutations, binary operand swaps, arithmetic operator toggles, and switch and if substitutions. Minor perturbations involve adding redundant statements, altering a few words in comments/snippets, or injecting unusable yet syntactically valid code. Sentence- and document-level perturbations include reordering sentences, manipulating keywords, and appending crafted sentences to influence model behavior. At the smallest granularity, character-level perturbations involve inserting or replacing characters, using invisible characters, homoglyphs (e.g., replacing "o" with "0"), reordering characters, or applying phonetic changes to mislead text models [S91, S92, S93, S94, S95, S96, S97, S98, S99]. All these perturbations aim to maintain plausibility to human readers while effectively compromising model robustness.

3D Point Clouds perturbation patterns operate directly on the coordinates (x, y, z) of points in the point cloud, slightly altering the perceived geometry of the object without changing its macroscopic shape. Some perturbations operate in feature space, affecting parameters such as distance, altitude, and azimuth alterations (up to 8°), as well as geometric transformations such as rotation, scaling, and translation. These changes are designed to be imperceptible to humans but sufficiently disruptive to fool deep learning models [S100, S101, S102, S103].

Node- and edge-level perturbations can be classified into two types: those that alter node or edge features without modifying the graph structure, and those that manipulate the structure itself. Node-level perturbations directly affect the features associated with individual nodes. For example, in datasets of academic papers, each node may represent a document characterized by keywords as attributes. An attacker can subtly replace meaningful words (e.g., "chemical", "david") with irrelevant or misleading ones. These modifications are often minimal but strategically crafted to mislead the model during classification or inference [S104, S105, S106, S107, S108, S109, S110].

Patch-level perturbation pattern involves inserting modified and localized patches into real images, designed to remain "natural" and not obvious to the human eye. Perturbations are often generated via GANs or guided optimizations and can be physically printed as stickers or posters to be applied to real-world objects (e.g., road signs). These patches are designed to withstand variations in distance, angle, light conditions, and motion. Effectiveness is very high, even in black-box conditions or dynamic environments. Graffiti-style stickers represent a particularly dangerous form of attack because they combine stealth with real-world effectiveness [S111, S112, S113, S114, S115, S116, S117, S118].

Waveform-level perturbations involve the addition of ambient noise or sudden sounds to the audio signal. They can be natural (traffic, rain, voice) or synthetic, and manifest themselves directly on the waveform. Perturbations can be both physical and digital and have been analyzed in several real-world scenarios, including living rooms, offices, airports, and shopping malls. Another critical aspect involves manipulating the volume and distance of the adversary signal, with measurements in dB SPL. Particular attention was paid to imperceptible noise. Even at low intensity (SNR $>$ 20dB, inaudible to the human ear), these noises can cause significant degradation in speech recognition or command systems, especially when the model is sensitive to acoustic or semantic context [S119, S120, S121, S122, S123].

Frame-level: The frame-level perturbation pattern identifies specific portions of the video sequence (entire frames or patches of frames) to be strategically modified to cause errors in the model, while keeping most of the content unchanged. The perturbations are then scattered in time, involving only a few selected frames, or concentrated in well-defined time windows through the use of temporal masks. The techniques described include advanced optimization strategies based on the L_1, L_2 norm, combined with the use of temporal masking to maximize the effectiveness of perturbations in both space and time. In other cases, the single frame is divided into blocks (patches) on which the attack is concentrated, allowing for a more targeted and less perceptible manipulation [S124, S125, S28].

Trojan trigger perturbation pattern refers to visible or invisible changes applied directly to the input during the training phase, to activate a malicious behavior in the model. The trigger acts as a "switch" that, when detected, induces the model to produce a specific output desired by the attacker, often without degrading performance on legitimate inputs. Triggers can vary in position, shape (square, logo, watermark), size (e.g. 1%–25% of the image surface), and transparency (e.g., 0%–70%) [S126, S127, S128, S129, S130, S131, S132, S133, S134, S135]. Perturbation techniques include manipulations such as overlaying images or parts of them (patch/crop/composite), as well as semantic patterns invisible to pixel-based controls. Another relevant category is audio perturbations [S136], where short sound signals (such as birds chirping or engine noises) act as triggers, inserted into the training data with variable poisoning percentages and with minimum trigger sizes. Finally, TrojanNet is implemented by inserting dedicated modules within the network, which activate backdoor behaviors only in the presence of specific patterns. This substructure works in parallel to the main model and acts as a classification shortcut: in the presence of the trigger, it can overwrite the original prediction, altering the behavior of the model without significantly changing its structure or performance on clean data [S137].

Feature-level perturbation pattern involves the direct manipulation of the numerical data that serve as inputs to machine learning models. One example is the perturbation of time series data points, where the perturbation ϵ is progressively increased, resulting in a significant degradation of performance in models [S138]. Another scenario concerns the creation of adversarial users, where the objective is to determine the minimal perturbation vector that, when added to the feature vector of a sample, causes a classifier to misclassify it without raising suspicion. In this case, a matrix of dummy users × items is generated so that the distribution of ratings or interactions remains as close as possible to that of real users. A further example is adversarial feature injection, in which perturbations are applied to 35 dynamic features, treating each network flow or connection as a numerical vector, much like an image is represented by a pixel matrix [S139, S140, S68].

Parameter and hyperparameter perturbation pattern refers to those attack techniques that act on the internal representations of the model, manipulating elements such as specific layers, feature maps, number of neurons with the aim of altering the behavior of the system. The focus is on how the model builds, transforms, and uses the latent features [S141, S142, S143, S144]. Another relevant type is represented by Backdoor/Physical attacks, based on the injection of backdoors into the model through weight manipulation, which are activated only under specific conditions, for example, after transfer learning. Another example concerns architectural perturbations, such as the intentional addition of malicious nodes and connections into the structure of the adversarial model, which alters the computation and induces divergent outputs. Finally, intentional modifications of optimization parameters emerge, which involve controlled variations of hyperparameters such as learning rate, weight decay, and the choice of the optimization algorithm.

Query-based perturbation pattern includes attacks that indirectly manipulate the model through a strategic sequence of queries. The goal is to infer information about the internal structure, weights, or training data of the model or data. Attackers can build surrogate models, reconstruct sensitive inputs via model inversion, or violate privacy by detecting whether an example belongs to the training set (membership inference). Unlike perturbations that act directly on the input or weights, these interferences rely on external interactions with the model, via APIs, predictions, or returned probabilities. The attack is particularly dangerous in black-box environments, where the only access channel is the model response [S145, S146, S147, S148, S149, S150, S151, S152].

B. *RQ_2: What is the relationship between the amount and type of perturbation and their impact?*

Upon thoroughly examining the existing literature concerning the impact of AAs on machine learning systems, to assess our RQ_2, our attention has shifted towards identifying the specific areas in which the impact manifests most critically. We aim to gain insight into the consequences and vulnerabilities exposed by these attacks, highlighting the dimensions where their effect is most profound. The main impact dimensions have been grouped into three categories:

Model performance degradation varies significantly depending on the type of perturbation applied. *Pixel-level* and *waveform-level* perturbations demonstrate a highly nonlinear relationship between the amount of perturbation and the performance degradation: even small changes (e.g., subtle changes in a single pixel or a slight audio noise) can cause a large performance degradation, with $L_0 = 2$ achieve an ASR of up to 100%, reducing key metrics such as AUC from 0.87 to 0.52 [S66], with particularly severe impacts in medical imaging models, highlighting the fragility of computer vision and speech recognition models. In the NLP domain, even minimal, semantics-preserving adversarial perturbations can significantly degrade model performance in both traditional NLP tasks [S94, S83, S82] and code-related tasks [S90, S96, S99]. These manipulations can lead to drastic reductions in performance metrics; for instance, *BLEU scores* can decrease by over 70% [S90]. Empirical results demonstrate reductions in *CodeBLEU* between 19.72% and 38.74% for models like CodeGPT, PLBART, and CodeT5 [S99]. Importantly, models that rely heavily on contextual information, such as LSTM- or Transformer-based architectures (e.g., CodeBERT), exhibit greater susceptibility compared to models incorporating structural information (e.g., GNN-based models or Rencos) [S90]. Specific attacks such as ALERT demonstrate adversarial training can later restore robustness, with CodeBERT and GraphCodeBERT showing post-adversarial tuning *accuracy* increases of 87.59% and 92.32%, respectively [S86]. In the physical context, slight *geometric transformations* are enough to achieve *ASR* up to 100% [S69]. Imperceptible perturbations applied to waveform patterns have been shown to achieve an *ASR* of 99.5% [S121]. While physically applied patch-level perturbations, such as camouflaged sticker attacks, have demonstrated a 100% *ASR* [S113]. Physical or digital *triggers*, even invisible ones, can activate unwanted behaviors while maintaining high performance on clean data (ASR = 100%), as demonstrated in TrojanNet, which remains inactive in the absence of the trigger [S137]. In the *binary-level* perturbations, the insertion of sequences from 500 to 20,000 bytes can generate *ASR*s between 74% and 99.5% [S65]. Even small textual changes, such as adding 5-10 keywords in a PDF [S67], can fool linear models such as SVM, highlighting their high sensitivity even to local perturbations. *Bytecode* or *feature* attacks significantly degrade the performance of malware detection systems: alterations affecting only 0.0004% of the features lead to errors in 63% of malicious samples [S77]. In the *3D point-level*, small perturbations ($\varepsilon_\infty = 0.18$) [S103, S100] or the addition of 20-60 spoofed points drastically reduce *accuracy* with *ASR* $\approx$ 75% [S102], especially on models pre-trained on complex datasets such as ModelNet40. In graphs, three structural changes significantly compromise performance [S104, S153]. *Query-based* perturbations do not directly degrade performance during the attack, but allow the construction of equivalent models for malicious uses. In some cases, reconstruction can introduce distortion without

TABLE II
DESCRIPTION OF PERTURBATION PATTERNS IN ADVERSARIAL ATTACKS

Perturbation Category	Perturbation Pattern		Perturbation Strength
Data Input	*Pixel-level*	single pixel changes; most influential pixel, jigsaw puzzles, color, texture, brightness, illumination, contrast, hue, saturation, Gaussian blur, Gaussian noise	L_p: p $\in$ [0, 1, 2, L_∞]
	Packet/Byte-level	appending bytes, modifying bytes	number of bytes
	Bytecode-level	feature mutation (permissions, intents, activities); combination of manifest and dex level changes;	number of injected gadgets; size of injected payload; number of bytecode transformations
	Word-level	synonym substitution (semantic/syntactic); sememe substitution; word deletion or addition (tokens, stopwords); token manipulation; code identifier renaming (in source code); inflectional perturbation (verb/noun tense or number); semantic-preserving modifications to code snippets; add invisible characters; homoglyph substitution; reordering characters; backspace injection; phonetic variation;	number of words per sentence; number of sememes; number of tokens; number of identifiers; number of chars per word;
	3D point clouds	deforming the coordinates of 3D points (e.g., altitude, azimuth, distance); transformations: drop, flip, rotate, scale, shear, translate	number 3d point cloud spoofed;
	Node/Edge-level	adding, deleting nodes or adding, deleting edges; node feature manipulation; injection of fake nodes; structure poisoning;	perturbation rate (%); perturbations per node (count); noise ratio (edges); node injection limit; misclassification threshold (Δ changes);
	Patch-level	small patches visually integrated into the context; graffiti-style stickers application; large size poster application	patch size (%)
	Geometric transformation	rotation, distortion, translation, shift, shearing, scaling, perspective, random resizing, random padding, cropping, overlaying	rotation (angle); perturb. area = (%);
	Waveform-level	addition of ambient noises: traffic, rain, engine, air conditioning; sudden sounds: ringtone, extraneous voice, ultrasonic sounds; overlapping speech or semantic interference (other speakers, similar sentences or semantic distractors)	SNR; noise level (dBSPL); percentage of noise; amplitude of waveform perturbation; distance (meters); frequency range (Hz)
	Frame-level	temporal sparse perturbation (e.g., perturb only one or few frames), patch-based perturbation (frame divided into patches), randomized mask, temporal window perturbation	number of frames; scaling factor; perturbation bound per frame; translation dx, dy (pixels); Gaussian noise
	Feature-level	numerical values of the time series; feature injection; ventor perturbation;	perturbation rate (%)
Model	*Parameter and hyper-parameter*	weight-space manipulation; model layer; feature maps; nmber of neurons;	weight, specific layers, feature maps, number of neurons
	Trigger-level	visual trigger insertion (such as tattoos, masks, logos, handwritten letters, alert icons, overlay images), physical (applied in the real world) or digital (inserted into the data during training)	perturbed image (%) of image; transparency (%); poisoning (%)
	Query-based	model extraction; model inversion; shadow model training; query-based reconstruction; membership inference;	number of queries

compromising the overall effectiveness of the attack. Furthermore, intensive use of APIs can compromise the operational efficiency of production systems.

Security and robustness threats. Several mitigation techniques have been proposed in the scientific literature, aiming to reduce the impact of adversarial perturbations and improve the robustness of models. These strategies include approaches based on adversarial training, dynamic perturbation detection, model distillation, randomization, and feature squeezing techniques, all designed to mitigate the effects of attacks and ensure more stable performance [S154, S155, S156, S157, S158, S159, S160]. However, despite the progress made, defenses still show several limitations. AAs pose significant threats to the security and robustness of machine learning models, exposing their vulnerability and limiting their resilience to attacks. In general, despite the progress, many defenses such as denoising [S36, S161], super-resolution, and adversarial training show only partial effectiveness, often at the expense of predictive quality [S32, S61]. Attacks based on sparsity or perceptual constraints evade active defenses [S39]. In particular, packet- or byte-level perturbations are difficult to detect and remain effective even with selected features [S131]. Multi-layered approaches (manifest + executable code) compromise robustness, rendering permission- or API-based defenses ineffective [S79]. In graphs, simple structural changes outperform non-optimized defenses, while approaches such as RGCN and Pro-GNN offer improvements [S110]. Geometric transformations represent a concrete threat to visual models, with high transferability between architectures, even if trained with defensive techniques (FGSM, PGD, C&W) [S71]. Bytecode-level perturbations have less impact on standard performance metrics, but can significantly alter the model's ability to distinguish between benign and malicious code, causing critical errors in security contexts. Physical attacks maintain high effectiveness in the real world, eluding conventional defenses [S126]. Invisible backdoors, activated by realistic inputs, can elude even advanced techniques, showing high transferability and confirmed effectiveness on real devices [S129]. Defenses such as data augmentation or spectral analysis are not effective. Trojan attacks represent a serious threat, with minimal alterations difficult to detect even with advanced structural analysis [S162]. Model extraction, inversion, and membership inference attacks undermine intellectual property and privacy [S145, S147, S146], facilitating future manipulation.

Severity and exploitability factors. The severity of an adversarial attack is determined by the potential damage it can cause to the application or the end user, while exploitability reflects the ease with which the attack can be executed, taking into account the necessary knowledge, the required computational resources, and the type of access to the model. The severity assessment considers several factors: complexity of the attack, level of access required, available mitigations, and impact on data integrity, confidentiality, and availability [34] [S95, S94, S139, S66, S151, S150, S147, S146, S145, S69, S38, S113]. In AML contexts, it is also essential to consider transferability, i.e., the ability of an attack to maintain its effectiveness on different models, thus increasing the danger of the threat [S82, S72, S25, S3, S4]. For example, the Drebin attack [S80] achieved a 99% evasion rate by simply adding a few features on average, demonstrating that minimal perturbations can have a very serious impact. Attacks characterized by very small perturbation values are particularly insidious, especially in areas such as autonomous driving, where they can hide critical elements such as pedestrians with invisible noise, seriously compromising safety [S52]. Similarly, TEXTBUGGER [S92] shows high severity thanks to a high success rate even with a low level of perturbation, exploiting subtle changes and expanding the attack surface to include stop words and non-keywords. In the audio domain, threats countered by AntiFake [S123] demonstrate the severity of real-world vulnerabilities: financial fraud, sensitive data theft, and bypassing voice authentication systems can be performed simply by using public or stolen audio. Stealth attacks with high transferability are among the most dangerous because they work on different models without the need for internal access (black-box), are effective even on complex models in production, and are difficult to detect, especially when localized in physical patches rather than distributed across the entire input.

In summary, the threat is particularly critical when the attack is able to compromise the security of the system with minimal and undetectable changes, while maintaining high transferability and operability in black-box scenarios, making it easily reproducible and applicable in real contexts, reinforcing the relevance of adversarial threats for the security and reliability of AI systems.

V. Threats to Validity

To limit the threat to *descriptive validity* [35], a data collection form was designed. Poor design or inaccurate recording could compromise its quality, which is why the form was carefully designed and collaboratively reviewed. To mitigate the threat to *theoretical validity*, we reviewed key articles and established a reference set via forward snowballing. We selected Scopus as the primary database for its broad coverage, and also included IEEE and ACM to ensure completeness. The search, conducted in March 2025, covers only the first two months of the year, which may affect representativeness. To reduce this risk, we included multiple venues and held regular review meetings. To reduce the threat to *interpretive validity*, expert researchers were involved in the process. However, since this step involves human judgment, the threat cannot be eliminated. To ensure *repeatability*, we describe in detail the process followed and the actions taken to reduce threats to validity, adopting existing approaches [36, 35]. All collected data are available in the replication package [1].

VI. Conclusion and Future Work

In this systematic literature review, we aimed to provide a comprehensive overview of the current state of research in AML. By analyzing 162 studies, we offered a structured synthesis of the field, shedding light on the various strategies adopted to compromise or defend ML models. Our main contributions include: (i) a thorough categorization and analysis of AAs and defense techniques across different application domains; (ii) an investigation of the perturbation patterns, including input-level and model-level perturbations, and their role in attack effectiveness; (iii) an evaluation of the impact of perturbation type and strength on the robustness and security of ML models; and (iv) the release of a detailed online appendix containing all study references, extraction criteria, and materials used to support further replication and extension. Looking ahead, we intend to develop a vulnerability assessment framework that integrates our perturbation pattern taxonomy as a basis for adversarial behavior detection. Moreover, we aim to explore the use of perturbation impact as a prioritization criterion to guide risk assessment processes in machine learning systems. This dual approach, focused on detection and prioritization, can support more proactive and structured security assessments in adversarial settings.

Acknowledgment

This work has been partially supported by the EMELIOT national research project, which has been funded by the MUR under the PRIN 2020 program (Contract 2020W3A5FY). This work was partially supported by project SERICS (PE00000014) under the NRRP MUR program funded by the EU - NGEU. This work was partially supported by project FAIR (PE0000013) under the NRRP MUR program funded by the EU - NGEU.

References

[1] Alexandra Sheykina. Replication package for exploring perturbation patterns and impact in aml. Available at https://github.com/sashasheykina/Adversarial-Machine-Lerning-SRL, 2025.

[2] Marco Barreno, Blaine Nelson, Russell Sears, Anthony D Joseph, and J Doug Tygar. Can machine learning be secure? In *Proceedings of the ACM Symposium on Information, computer and communications security*, pages 16–25, 2006.

[3] Qiang Liu, Pan Li, Wentao Zhao, Wei Cai, Shui Yu, and Victor CM Leung. A survey on security threats and defensive techniques of machine learning: A data driven view. *IEEE access*, 6:12103–12117, 2018.

[4] Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018.

[5] Naveed Akhtar, Ajmal Mian, Navid Kardan, and Mubarak Shah. Advances in adversarial attacks and defenses in computer vision: A survey. *IEEE Access*, 9:155161–155196, 2021.

[6] Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pages 2154–2156, 2018.

[7] Wei Emma Zhang, Quan Z Sheng, Ahoud Alhazmi, and Chenliang Li. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology*, 11(3), 2020.

[8] Han Xu, Yao Ma, Hao-Chen Liu, Debayan Deb, Hui Liu, Ji-Liang Tang, and Anil K Jain. Adversarial attacks and defenses in images, graphs and text: A review. *International Journal of Automation and Computing*, 17:151–178, 2020.

[9] Wei Jin, Yaxing Li, Han Xu, Yiqi Wang, Shuiwang Ji, Charu Aggarwal, and Jiliang Tang. Adversarial attacks and defenses on graphs. *ACM SIGKDD Explorations Newsletter*, 22(2), 2021.

[10] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 30(9), 2019.

[11] Samuel Henrique Silva and Peyman Najafirad. Opportunities and challenges in deep learning adversarial robustness: A survey. *arXiv preprint arXiv:2007.00753*, 2020.

[12] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. A survey on adversarial attacks and defences. *CAAI Transactions on Intelligence Technology*, 6(1), 2021.

[13] Fatemeh Vakhshiteh, Ahmad Nickabadi, and Raghavendra Ramachandra. Adversarial attacks against face recognition: A comprehensive study. *IEEE Access*, 9:92735–92756, 2021.

[14] Alex Serban, Erik Poll, and Joost Visser. Adversarial examples on object recognition: A comprehensive survey. *ACM Computing Surveys*, 53(3), 2020.

[15] Fatimah Aloraini, Amir Javed, Omer Rana, and Pete Burnap. Adversarial machine learning in iot from an insider point of view. *Journal of Information Security and Applications*, 70:103341, 2022.

[16] Nikolaos Pitropakis, Emmanouil Panaousis, Thanassis Giannetsos, Eleftherios Anastasiadis, and George Loukas. A taxonomy and survey of attacks against machine learning. *Computer Science Review*, 34:100199, 2019.

[17] Kshitiz Aryal, Maanak Gupta, and Mahmoud Abdelsalam. A survey on adversarial attacks for malware analysis. *arXiv preprint arXiv:2111.08223*, 2021.

[18] Inaam Ilahi, Muhammad Usama, Junaid Qadir, Muhammad Umar Janjua, Ala Al-Fuqaha, Dinh Thai Hoang, and Dusit Niyato. Challenges and countermeasures for adversarial attacks on deep reinforcement learning. *IEEE Transactions on Artificial Intelligence*, 3(2), 2021.

[19] Yue Liu, Chakkrit Tantithamthavorn, Li Li, and Yepang Liu. Deep learning for android malware defenses: a systematic literature review. *ACM Computing Surveys*, 55(8), 2022.

[20] Zhibo Wang, Jingjing Ma, Xue Wang, Jiahui Hu, Zhan Qin, and Kui Ren. Threats to training: A survey of poisoning attacks and defenses on machine learning systems. *ACM Computing Surveys*, 55(7), 2022.

[21] Antonio Emanuele Cinà, Kathrin Grosse, Ambra Demontis, Sebastiano Vascon, Werner Zellinger, Bernhard A Moser, Alina Oprea, Battista Biggio, Marcello Pelillo, and Fabio Roli. Wild patterns reloaded: A survey of machine learning security against training data poisoning. *ACM Computing Surveys*, 55(13s), 2023.

[22] Huoyuan Dong, Jialiang Dong, Shuai Yuan, and Zhitao Guan. Adversarial attack and defense on natural language processing in deep learning: A survey and perspective. In *International Conference on Machine Learning for Cyber Security*, pages 409–424, 2022.

[23] Xiang Ling, Lingfei Wu, Jiangyu Zhang, Zhenqing Qu, Wei Deng, Xiang Chen, Yaguan Qian, Chunming Wu, Shouling Ji, Tianyue Luo, et al. Adversarial attacks against windows pe malware detection: A survey of the state-of-the-art. *Computers & Security*, page 103134, 2023.

[24] Antonio Emanuele Cinà, Kathrin Grosse, Ambra Demontis, Battista Biggio, Fabio Roli, and Marcello Pelillo. Machine learning security against data poisoning: Are we there yet? *arXiv preprint arXiv:2204.05986*, 2022.

[25] Ahmed Aldahdooh, Wassim Hamidouche, Sid Ahmed Fezza, and Olivier Déforges. Adversarial example detection for dnn models: A review and experimental comparison. *Artificial Intelligence Review*, 55(6), 2022.

[26] Mingfu Xue, Chengxiang Yuan, Heyi Wu, Yushu Zhang, and Weiqiang Liu. Machine learning security: Threats, countermeasures, and evaluations. *IEEE Access*, 8:74720–74742, 2020.

[27] Tong Chen, Jiqiang Liu, Yingxiao Xiang, Wenjia Niu, Endong Tong, and Zhen Han. Adversarial attack and defense in reinforcement learning-from ai security view. *Cybersecurity*, 2:1–22, 2019.

[28] Shilin Qiu, Qihe Liu, Shijie Zhou, and Chunjiang Wu. Review of artificial intelligence adversarial attack and defense technologies. *Applied Sciences*, 9(5), 2019.

[29] Weimin Zhao, Sanaa Alwidian, and Qusay H Mahmoud.

Adversarial training methods for deep learning: A systematic review. *Algorithms*, 15:283, 2022.

[30] Staffs Keele et al. Guidelines for performing systematic literature reviews in software engineering, 2007.

[31] Barbara Kitchenham. Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004):1–26, 2004.

[32] Victor R Basili1 Gianluigi Caldiera and H Dieter Rombach. The goal question metric approach. *Encyclopedia of software engineering*, pages 528–532, 1994.

[33] Claes Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, pages 1–10, 2014.

[34] NIST. Common vulnerability scoring system. Available at https://nvd.nist.gov/vuln-metrics/cvss, 2025.

[35] Kai Petersen, Robert Feldt, Shahid Mujtaba, and Michael Mattsson. Systematic mapping studies in software engineering. In *12th International Conference on Evaluation and Assessment in Software Engineering*, pages 1–10, 2008.

[36] Barbara Kitchenham, O Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman. Systematic literature reviews in software engineering–a systematic literature review. *Information and software technology*, 51(1), 2009.

SRL'S STUDIES

[S1] Jérôme Rony, Luiz G Hafemann, Luiz S Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric Granger. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4322–4330, 2019.

[S2] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy, EuroS&P*, pages 372–387, 2016.

[S3] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *5th International Conference on Learning Representations*, 2017.

[S4] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017.

[S5] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773, 2017.

[S6] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations*, 2014.

[S7] Jianbo Chen, Michael I Jordan, and Martin J Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. In *IEEE symposium on security and privacy*, pages 1277–1294, 2020.

[S8] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *Advances in neural information processing systems*, 33:1633–1645, 2020.

[S9] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283, 2018.

[S10] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. *Advances in neural information processing systems*, 31, 2018.

[S11] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*, pages 2137–2146, 2018.

[S12] Sandy H. Huang, Nicolas Papernot, Ian J. Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. In *5th International Conference on Learning Representations*, 2017.

[S13] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations*, 2018.

[S14] Xiaojun Jia, Xingxing Wei, Xiaochun Cao, and Hassan Foroosh. Comdefend: An efficient image compression model to defend adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6084–6092, 2019.

[S15] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[S16] Bo Sun, Nian-hsuan Tsai, Fangchen Liu, Ronald Yu, and Hao Su. Adversarial defense by stratified convolutional sparse coding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11447–11456, 2019.

[S17] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216, 2020.

[S18] Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. In *7th International Conference on Learning Representations*, 2019.

[S19] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations*, 2015.

[S20] Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. 2018.

[S21] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

[S22] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE symposium on security and privacy*, pages 39–57, 2017.

[S23] Ranjie Duan, Xingjun Ma, Yisen Wang, James Bailey, A Kai Qin, and Yun Yang. Adversarial camouflage: Hiding physical-world attacks with natural styles. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1000–1008, 2020.

[S24] Minhao Cheng, Simranjit Singh, Patrick H. Chen, Pin-Yu Chen, Sijia Liu, and Cho-Jui Hsieh. Sign-opt: A query-efficient hard-label adversarial attack. In *8th International Conference on Learning Representations*, 2020.

[S25] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[S26] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5), 2019.

[S27] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L. Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[S28] Paarth Neekhara, Brian Dolhansky, Joanna Bitton, and Cristian Canton Ferrer. Adversarial threats to deepfake detection: A practical perspective. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 923–932, 2021.

[S29] Xingjun Ma, Yuhao Niu, Lin Gu, Yisen Wang, Yitian Zhao, James Bailey, and Feng Lu. Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, 110:107332, 2021.

[S30] Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. Admix: Enhancing the transferability of adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16158–16167, 2021.

[S31] Anand Bhattad, Min Jin Chong, Kaizhao Liang, Bo Li, and David A. Forsyth. Unrestricted adversarial examples via semantic manipulation. In *8th International Conference on Learning Representations*, 2020.

[S32] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International conference on learning representations*, 2019.

[S33] Xiaoyi Dong, Dongdong Chen, Jianmin Bao, Chuan Qin, Lu Yuan, Weiming Zhang, Nenghai Yu, and Dong Chen. Greedyfool: Distortion-aware sparse adversarial attack. *Advances in Neural Information Processing Systems*, 33:11226–11236, 2020.

[S34] Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*, pages 2196–2205, 2020.

[S35] Qizhang Li, Yiwen Guo, and Hao Chen. Practical no-box adversarial attacks against dnns. *Advances in Neural Information Processing Systems*, 33:12849–12860, 2020.

[S36] Aamir Mustafa, Salman H Khan, Munawar Hayat, Jianbing Shen, and Ling Shao. Image super-resolution as a defense against adversarial attacks. *IEEE Transactions on Image Processing*, 29:1711–1724, 2019.

[S37] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. A self-supervised approach for adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 262–271, 2020.

[S38] Yinpeng Dong, Hang Su, Baoyuan Wu, Zhifeng Li, Wei Liu, Tong Zhang, and Jun Zhu. Efficient decision-based black-box adversarial attacks on face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7714–7722, 2019.

[S39] Francesco Croce and Matthias Hein. Sparse and imperceivable adversarial attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4724–4732, 2019.

[S40] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 501–509, 2019.

[S41] Yandong Li, Lijun Li, Liqiang Wang, Tong Zhang, and Boqing Gong. Nattack: Learning the distributions of adversarial examples for an improved black-box attack on deep neural networks. In *International Conference on Machine Learning*, pages 3866–3876, 2019.

[S42] Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *International conference on learning representations*, 2018.

[S43] Maksym Andriushchenko, Francesco Croce, Nicolas

Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European conference on computer vision*, pages 484–501. Springer, 2020.

[S44] Zhengyuan Jiang, Jinghuai Zhang, and Neil Zhenqiang Gong. Evading watermark based detection of ai-generated content. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1168–1181, 2023.

[S45] Junhao Dong, Seyed-Mohsen Moosavi-Dezfooli, Jianhuang Lai, and Xiaohua Xie. The enemy of my enemy is my friend: Exploring inverse adversaries for improving adversarial training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24678–24687, 2023.

[S46] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Siwei Li, Li Chen, Michael E Kounavis, and Duen Horng Chau. Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 196–204, 2018.

[S47] Moustafa Alzantot, Yash Sharma, Supriyo Chakraborty, Huan Zhang, Cho-Jui Hsieh, and Mani B Srivastava. Genattack: Practical black-box attacks with gradient-free optimization. In *Proceedings of the genetic and Evolutionary Computation Conference*, pages 1111–1119, 2019.

[S48] Jinghui Chen and Quanquan Gu. Rays: A ray searching method for hard-label adversarial attack. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1739–1747, 2020.

[S49] Shumeet Baluja and Ian Fischer. Learning to attack: Adversarial transformation networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[S50] Guneet S. Dhillon, Kamyar Azizzadenesheli, Zachary C. Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, and Animashree Anandkumar. Stochastic activation pruning for robust adversarial defense. In *6th International Conference on Learning Representations*, 2018.

[S51] Pin-Yu Chen, Yash Sharma, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh. Ead: elastic-net attacks to deep neural networks via adversarial examples. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[S52] Jan Hendrik Metzen, Mummadi Chaithanya Kumar, Thomas Brox, and Volker Fischer. Universal adversarial perturbations against semantic image segmentation. In *Proceedings of the IEEE International Conference on Computer vision*, pages 2755–2764, 2017.

[S53] Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 742–749, 2019.

[S54] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *ICLR*, 2018.

[S55] Gaurav Goswami, Akshay Agarwal, Nalini Ratha, Richa Singh, and Mayank Vatsa. Detecting and mitigating adversarial perturbations for robust face recognition. *International Journal of Computer Vision*, 127:719–742, 2019.

[S56] Dongxian Wu, Yisen Wang, Shu-Tao Xia, James Bailey, and Xingjun Ma. Skip connections matter: On the transferability of adversarial examples generated with resnets. *arXiv preprint arXiv:2002.05990*, 2020.

[S57] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3681–3688, 2019.

[S58] Xiaoyu Cao and Neil Zhenqiang Gong. Mitigating evasion attacks to deep neural networks via region-based classification. In *Proceedings of the 33rd annual computer security applications conference*, pages 278–287, 2017.

[S59] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. In *6th International Conference on Learning Representations*, 2018.

[S60] Dongyu Meng and Hao Chen. Magnet: a two-pronged defense against adversarial examples. In *Proceedings of the ACM SIGSAC conference on computer and communications security*, pages 135–147, 2017.

[S61] Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. Improving the generalization of adversarial training with domain adaptation. In *7th International Conference on Learning Representations*, 2019.

[S62] Zeyu Dai, Shengcai Liu, Qing Li, and Ke Tang. Saliency attack: Towards imperceptible black-box adversarial attack. *ACM Transactions on Intelligent Systems and Technology*, 14(3):1–20, 2023.

[S63] Jingyi Wang, Guoliang Dong, Jun Sun, Xinyu Wang, and Peixin Zhang. Adversarial sample detection for deep neural network through model mutation testing. In *IEEE/ACM 41st International Conference on Software Engineering*, pages 1245–1256, 2019.

[S64] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1778–1787, 2018.

[S65] Bingcai Chen, Zhongru Ren, Chao Yu, Iftikhar Hussain, and Jintao Liu. Adversarial examples for cnn-based malware detectors. *IEEE Access*, 7:54360–54371, 2019.

[S66] Alesia Chernikova and Alina Oprea. Fence: Feasible evasion attacks on neural networks in constrained environments. *ACM Transactions on Privacy and Security*, 25(4), 2022.

[S67] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Machine Learning and Knowledge Discovery in Databases: European Conference*, pages 387–402, 2013.

[S68] Khushnaseeb Roshan, Aasim Zafar, and Shiekh Burhan Ul Haque. Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system. *Computer Communications*, 218:97–113, 2024.

[S69] Bowen Zhang, Benedetta Tondi, and Mauro Barni. Adversarial examples for replay attacks against cnn-based face recognition with anti-spoofing capability. *Comput. Vis. Image Underst.*, 197-198:102988, 2020.

[S70] Shixin Tian, Guolei Yang, and Ying Cai. Detecting adversarial examples through image transformation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[S71] Chaowei Xiao, Jun-Yan Zhu, Bo Li, Warren He, Mingyan Liu, and Dawn Song. Spatially transformed adversarial examples. In *6th International Conference on Learning Representations*, 2018.

[S72] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. *arXiv preprint arXiv:1908.06281*, 2019.

[S73] Changhao Shi, Chester Holtz, and Gal Mishne. Online adversarial purification based on self-supervised learning. In *9th International Conference on Learning Representations*, 2021.

[S74] Yue Zhao, Hong Zhu, Ruigang Liang, Qintao Shen, Shengzhi Zhang, and Kai Chen. Seeing isn't believing: Towards more robust adversarial attack against real world object detectors. In *Proceedings of the ACM SIGSAC conference on computer and communications security*, pages 1989–2004, 2019.

[S75] Shang-Tse Chen, Cory Cornelius, Jason Martin, and Duen Horng Chau. Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2018, Dublin, Ireland, September 10–14, 2018, Proceedings, Part I 18*, pages 52–68. Springer, 2019.

[S76] Rima Alaifari, Giovanni S. Alberti, and Tandri Gauksson. Adef: an iterative algorithm to construct adversarial deformations. In *7th International Conference on Learning Representations*, 2019.

[S77] Kathrin Grosse, Nicolas Papernot, Praveen Manoharan, Michael Backes, and Patrick McDaniel. Adversarial examples for malware detection. In *22nd European Symposium on Research in Computer Security*, pages 62–79, 2017.

[S78] Hamid Bostani and Veelasha Moonsamy. Evadedroid: A practical evasion attack on machine learning for black-box android malware detection. *Computers & Security*, 139:103676, 2024.

[S79] Wei Yang, Deguang Kong, Tao Xie, and Carl A Gunter. Malware detection in adversarial settings: Exploiting feature evolutions and confusions in android apps. In *Proceedings of the 33rd Annual Computer Security Applications Conference*, pages 288–302, 2017.

[S80] Xiao Chen, Chaoran Li, Derui Wang, Sheng Wen, Jun Zhang, Surya Nepal, Yang Xiang, and Kui Ren. Android hiv: A study of repackaging malware for evading machine-learning detection. *IEEE Transactions on Information Forensics and Security*, 15:987–1001, 2019.

[S81] Huangzhao Zhang, Zhuo Li, Ge Li, Lei Ma, Yang Liu, and Zhi Jin. Generating adversarial examples for holding robustness of source code processing models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 1169–1176, 2020.

[S82] Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080, 2020.

[S83] Minhao Cheng, Jinfeng Yi, Pin-Yu Chen, Huan Zhang, and Cho-Jui Hsieh. Seq2sick: Evaluating the robustness of sequence-to-sequence models with adversarial examples. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 2020.

[S84] Yu-Lun Hsieh, Minhao Cheng, Da-Cheng Juan, Wei Wei, Wen-Lian Hsu, and Cho-Jui Hsieh. On the robustness of self-attentive models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1520–1529, 2019.

[S85] Ivan Fursov, Matvey Morozov, Nina Kaploukhaya, Elizaveta Kovtun, Rodrigo Rivera-Castro, Gleb Gusev, Dmitry Babaev, Ivan Kireev, Alexey Zaytsev, and Evgeny Burnaev. Adversarial attacks on deep models for financial transaction records. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2868–2878, 2021.

[S86] Zhou Yang, Jieke Shi, Junda He, and David Lo. Natural attack for pre-trained models of code. In *Proceedings of the 44th International Conference on Software Engineering*, pages 1482–1493, 2022.

[S87] Nilesh Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 99–108, 2004.

[S88] Zhihong Shao, Zhongqin Wu, and Minlie Huang. Advexpander: Generating natural language adversarial examples by expanding text. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1184–

1196, 2021.

[S89] Noam Yefet, Uri Alon, and Eran Yahav. Adversarial examples for models of code. *Proceedings of the ACM on Programming Languages*, 4(OOPSLA):1–30, 2020.

[S90] Yu Zhou, Xiaoqing Zhang, Juanjuan Shen, Tingting Han, Taolue Chen, and Harald Gall. Adversarial robustness of deep code comment generation. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 31(4):1–30, 2022.

[S91] Moustapha Cisse, Yossi Adi, Natalia Neverova, and Joseph Keshet. Houdini: Fooling deep structured prediction models. *arXiv preprint arXiv:1707.05373*, 2017.

[S92] Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. Textbugger: Generating adversarial text against real-world applications. *arXiv preprint arXiv:1812.05271*, 2018.

[S93] Ashish Bajaj and Dinesh Kumar Vishwakarma. Inflect-text: a novel mechanism to evade neural text classifiers by leveraging word inflectional perturbations. *International Journal of Information Security*, 24(1):70, 2025.

[S94] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025, 2020.

[S95] Nicholas Boucher, Ilia Shumailov, Ross Anderson, and Nicolas Papernot. Bad characters: Imperceptible nlp attacks. In *IEEE Symposium on Security and Privacy*, pages 1987–2004, 2022.

[S96] Xiaohu Du, Ming Wen, Zichao Wei, Shangwen Wang, and Hai Jin. An extensive study on adversarial attack against pre-trained models of code. In *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 489–501, 2023.

[S97] Ashish Bajaj and Dinesh Kumar Vishwakarma. Homochar: A novel adversarial attack framework for exposing the vulnerability of text based neural sentiment classifiers. *Engineering Applications of Artificial Intelligence*, 126:106815, 2023.

[S98] Fengjuan Gao, Yu Wang, and Ke Wang. Discrete adversarial attack to models of code. *Proceedings of the ACM on Programming Languages*, 7(PLDI):172–195, 2023.

[S99] Guang Yang, Yu Zhou, Wenhua Yang, Tao Yue, Xiang Chen, and Taolue Chen. How important are good method names in neural code generation? a model robustness perspective. *ACM Transactions on Software Engineering and Methodology*, 33(3):1–35, 2024.

[S100] Abdullah Hamdi, Sara Rojas, Ali Thabet, and Bernard Ghanem. Advpc: Transferable adversarial perturbations on 3d point clouds. In *Computer Vision–ECCV: 16th European Conference*, pages 241–257, 2020.

[S101] Hang Zhou, Dongdong Chen, Jing Liao, Kejiang Chen, Xiaoyi Dong, Kunlin Liu, Weiming Zhang, Gang Hua, and Nenghai Yu. Lg-gan: Label guided adversarial network for flexible targeted attack of point cloud based deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10356–10365, 2020.

[S102] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pages 2267–2281, 2019.

[S103] Fulan Qian, Yuanjun Zou, Mengyao Xu, Xuejun Zhang, Chonghao Zhang, Chenchu Xu, and Hai Chen. A comprehensive understanding of the impact of data augmentation on the transferability of 3d adversarial examples. *ACM Transactions on Knowledge Discovery from Data*, 19(2):1–41, 2025.

[S104] Dingyuan Zhu, Ziwei Zhang, Peng Cui, and Wenwu Zhu. Robust graph convolutional networks against adversarial attacks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1399–1407, 2019.

[S105] Negin Entezari, Saba A Al-Sayouri, Amirali Darvishzadeh, and Evangelos E Papalexakis. All you need is low (rank) defending against adversarial attacks on graphs. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 169–177, 2020.

[S106] Xu Zou, Qinkai Zheng, Yuxiao Dong, Xinyu Guan, Evgeny Kharlamov, Jialiang Lu, and Jie Tang. Tdgia: Effective injection attacks on graph neural networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2461–2471, 2021.

[S107] Daniel Zügner, Oliver Borchert, Amir Akbarnejad, and Stephan Günnemann. Adversarial attacks on graph neural networks: Perturbations and their patterns. *ACM Transactions on Knowledge Discovery from Data*, 14(5), 2020.

[S108] Sofiane Ennadir, Yassine Abbahaddou, Johannes F Lutzeyer, Michalis Vazirgiannis, and Henrik Boström. A simple and yet fairly effective defense for graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21063–21071, 2024.

[S109] Daniel Zügner, Amir Akbarnejad, and Stephan Günnemann. Adversarial attacks on neural networks for graph data. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2847–2856, 2018.

[S110] Wei Jin, Yao Ma, Xiaorui Liu, Xianfeng Tang, Suhang Wang, and Jiliang Tang. Graph structure learning for robust graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 66–74, 2020.

[S111] Aishan Liu, Xianglong Liu, Jiaxin Fan, Yuqing Ma, Anlan Zhang, Huiyuan Xie, and Dacheng Tao. Perceptual-sensitive gan for generating adversarial patches. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 2019.

[S112] Dawn Song, Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, and Tadayoshi Kohno. Physical adversarial examples for object detectors. In *12th USENIX workshop on offensive technologies (WOOT 18)*, 2018.

[S113] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[S114] Yu Ran, Weijia Wang, Mingjie Li, Lin-Cheng Li, Yuan-Gen Wang, and Jin Li. Cross-shaped adversarial patch attack. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(4):2289–2303, 2023.

[S115] Xingxing Wei, Ying Guo, and Jie Yu. Adversarial sticker: A stealthy attack method in the physical world. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):2711–2725, 2022.

[S116] Ziqi Zhou, Shengshan Hu, Minghui Li, Hangtao Zhang, Yechao Zhang, and Hai Jin. Advclip: Downstream-agnostic adversarial examples in multimodal contrastive learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6311–6320, 2023.

[S117] Xin Zheng, Yanbo Fan, Baoyuan Wu, Yong Zhang, Jue Wang, and Shirui Pan. Robust physical-world attacks on face recognition. *Pattern Recognition*, 133:109009, 2023.

[S118] Zhaoyu Chen, Bo Li, Shuang Wu, Shouhong Ding, and Wenqiang Zhang. Query-efficient decision-based black-box patch attack. *IEEE Transactions on Information Forensics and Security*, 18:5522–5536, 2023.

[S119] Hyun Kwon, Yongchul Kim, Hyunsoo Yoon, and Daeseon Choi. Selective audio adversarial example in evasion attack on speech recognition system. *IEEE Transactions on Information Forensics and Security*, 15:526–538, 2019.

[S120] Zhuohang Li, Yi Wu, Jian Liu, Yingying Chen, and Bo Yuan. Advpulse: Universal, synchronization-free, and targeted audio adversarial attacks via subsecond perturbations. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pages 1121–1134, 2020.

[S121] Tianyu Du, Shouling Ji, Jinfeng Li, Qinchen Gu, Ting Wang, and Raheem Beyah. Sirenattack: Generating adversarial audio for end-to-end acoustic systems. In *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security*, pages 357–369, 2020.

[S122] Guoming Zhang, Chen Yan, Xiaoyu Ji, Tianchen Zhang, Taimin Zhang, and Wenyuan Xu. Dolphinattack: Inaudible voice commands. In *Proceedings of the ACM SIGSAC conference on computer and communications security*, pages 103–117, 2017.

[S123] Zhiyuan Yu, Shixuan Zhai, and Ning Zhang. Antifake: Using adversarial audio to prevent unauthorized speech synthesis. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 460–474, 2023.

[S124] Linxi Jiang, Xingjun Ma, Shaoxiang Chen, James Bailey, and Yu-Gang Jiang. Black-box adversarial attacks on video recognition models. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 864–872, 2019.

[S125] Xingxing Wei, Jun Zhu, Sha Yuan, and Hang Su. Sparse adversarial perturbations for videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8973–8980, 2019.

[S126] Jiakai Wang, Aishan Liu, Zixin Yin, Shunchang Liu, Shiyu Tang, and Xianglong Liu. Dual attention suppression attack: Generate adversarial camouflage in physical world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8565–8574, 2021.

[S127] Junyu Lin, Lei Xu, Yingqi Liu, and Xiangyu Zhang. Composite backdoor attack for deep neural network by mixing existing benign features. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pages 113–131, 2020.

[S128] Bao Gia Doan, Ehsan Abbasnejad, and Damith C Ranasinghe. Februus: Input purification defense against trojan attacks on deep neural network systems. In *Annual computer security applications conference*, pages 897–912, 2020.

[S129] Yuanchun Li, Jiayi Hua, Haoyu Wang, Chunyang Chen, and Yunxin Liu. Deeppayload: Black-box backdoor attack on deep learning models through neural payload injection. In *IEEE/ACM 43rd International Conference on Software Engineering*, pages 263–274, 2021.

[S130] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pages 113–125, 2019.

[S131] Yuanshun Yao, Huiying Li, Haitao Zheng, and Ben Y Zhao. Latent backdoor attacks on deep neural networks. In *Proceedings of the ACM SIGSAC conference on computer and communications security*, pages 2041–2055, 2019.

[S132] Yingqi Liu, Wen-Chuan Lee, Guanhong Tao, Shiqing Ma, Yousra Aafer, and Xiangyu Zhang. Abs: Scanning neural networks for back-doors by artificial brain stimulation. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pages 1265–1282, 2019.

[S133] Shihao Zhao, Xingjun Ma, Xiang Zheng, James Bailey, Jingjing Chen, and Yu-Gang Jiang. Clean-label backdoor attacks on video recognition models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14443–14452, 2020.

[S134] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. In *25th Annual Network And Distributed System Security Symposium*. Internet Soc, 2018.

[S135] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *IEEE Symposium on Security and Privacy*, pages 707–723, 2019.

[S136] Cong Shi, Tianfang Zhang, Zhuohang Li, Huy Phan, Tianming Zhao, Yan Wang, Jian Liu, Bo Yuan, and Yingying Chen. Audio-domain position-independent backdoor attack via unnoticeable triggers. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, pages 583–595, 2022.

[S137] Ruixiang Tang, Mengnan Du, Ninghao Liu, Fan Yang, and Xia Hu. An embarrassingly simple approach for trojan attack in deep neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 218–228, 2020.

[S138] Tao Wu, Xuechun Wang, Shaojie Qiao, Xingping Xian, Yanbing Liu, and Liang Zhang. Small perturbations are enough: Adversarial attacks on time series prediction. *Information Sciences*, 587:794–812, 2022.

[S139] Christakopoulou Konstantina and Arindam Banerjee. Adversarial attacks on an oblivious recommender. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 322–330, 2019.

[S140] Xinwei Yuan, Shu Han, Wei Huang, Hongliang Ye, Xianglong Kong, and Fan Zhang. A simple framework to enhance the adversarial robustness of deep learning-based intrusion detection system. *Computers & Security*, 137:103644, 2024.

[S141] Hyun Kwon and Jang-Woon Baek. Targeted discrepancy attacks: Crafting selective adversarial examples in graph neural networks. *IEEE Access*, 2025.

[S142] Nathan Inkawhich, Wei Wen, Hai Helen Li, and Yiran Chen. Feature space perturbations yield more transferable adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7066–7074, 2019.

[S143] Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4733–4742, 2019.

[S144] Ren Pang, Hua Shen, Xinyang Zhang, Shouling Ji, Yevgeniy Vorobeychik, Xiapu Luo, Alex Liu, and Ting Wang. A tale of evil twins: Adversarial inputs versus poisoned models. In *Proceedings of the ACM SIGSAC conference on computer and communications security*, pages 85–99, 2020.

[S145] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1322–1333, 2015.

[S146] 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.

[S147] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE symposium on security and privacy*, pages 3–18, 2017.

[S148] Xing Hu, Ling Liang, Shuangchen Li, Lei Deng, Pengfei Zuo, Yu Ji, Xinfeng Xie, Yufei Ding, Chang Liu, Timothy Sherwood, et al. Deepsniffer: A dnn model extraction framework based on learning architectural hints. In *Proceedings of the 25th International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 385–399, 2020.

[S149] Baolin Zheng, Peipei Jiang, Qian Wang, Qi Li, Chao Shen, Cong Wang, Yunjie Ge, Qingyang Teng, and Shenyi Zhang. Black-box adversarial attacks on commercial speech platforms with minimal information. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pages 86–107, 2021.

[S150] Zheng Li and Yang Zhang. Membership leakage in label-only exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 880–895, 2021.

[S151] Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. Memguard: Defending against black-box membership inference attacks via adversarial examples. In *Proceedings of the ACM SIGSAC conference on computer and communications security*, pages 259–274, 2019.

[S152] Mingyi Zhou, Jing Wu, Yipeng Liu, Shuaicheng Liu, and Ce Zhu. Dast: Data-free substitute training for adversarial attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 234–243, 2020.

[S153] Xianfeng Tang, Yandong Li, Yiwei Sun, Huaxiu Yao, Prasenjit Mitra, and Suhang Wang. Transferring robustness for graph neural network against poisoning attacks. In *Proceedings of the 13th International Conference on web search and Data Mining*, pages 600–608, 2020.

[S154] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian J. Goodfellow, Dan Boneh, and Patrick D. McDaniel. Ensemble adversarial training: Attacks and defenses. In *6th International Conference on Learning Representations*, 2018.

[S155] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. In *5th International Conference on Learning Representations*, 2017.

[S156] Zhihao Zheng and Pengyu Hong. Robust detection of adversarial attacks by modeling the intrinsic properties of deep neural networks. *Advances in neural information processing systems*, 31, 2018.

[S157] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *IEEE symposium on security and privacy*, pages 582–597, 2016.

[S158] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan L. Yuille. Mitigating adversarial effects through randomization. In *6th International Conference on Learning Representations*. OpenReview.net, 2018.

[S159] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *25th Annual Network and Distributed System Security Symposium*, 2018.

[S160] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8), 2018.

[S161] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning*, pages 4970–4979, 2019.

[S162] Cheng-Hsin Weng, Yan-Ting Lee, and Shan-Hung Brandon Wu. On the trade-off between adversarial and backdoor robustness. *Advances in Neural Information Processing Systems*, 33:11973–11983, 2020.