Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 4907754, 22 pages
https://doi.org/10.1155/2021/4907754

Review Article

A Survey on Adversarial Attack in the Age of Artificial Intelligence

Zixiao Kong,[1] Jingfeng Xue,[1] Yong Wang ,[1] Lu Huang,[1] Zequn Niu,[1] and Feng Li[2]

[1] *School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China*
[2] *Shandong Cloudsky Security Technology Co., Ltd, Jinan, China*

Correspondence should be addressed to Yong Wang; wangyong@bit.edu.cn

Received 9 April 2021; Revised 24 May 2021; Accepted 11 June 2021; Published 22 June 2021

Academic Editor: Weizhi Meng

With the rapid evolution of the Internet, the application of artificial intelligence fields is more and more extensive, and the era of AI has come. At the same time, adversarial attacks in the AI field are also frequent. Therefore, the research into adversarial attack security is extremely urgent. An increasing number of researchers are working in this field. We provide a comprehensive review of the theories and methods that enable researchers to enter the field of adversarial attack. This article is according to the "Why? → What? → How?" research line for elaboration. Firstly, we explain the significance of adversarial attack. Then, we introduce the concepts, types, and hazards of adversarial attack. Finally, we review the typical attack algorithms and defense techniques in each application area. Facing the increasingly complex neural network model, this paper focuses on the fields of image, text, and malicious code and focuses on the adversarial attack classifications and methods of these three data types, so that researchers can quickly find their own type of study. At the end of this review, we also raised some discussions and open issues and compared them with other similar reviews.

1. Introduction

In the age of the Internet, with the accumulation of large amounts of data, the evolution of computing power, and the constant innovation and evolution of machine learning methods and frameworks, artificial intelligence (AI) technologies such as image recognition, machine translation, and autonomous driving have been widely deployed and widely applied all over the world [1]. Artificial intelligence is marching towards historic moments for mankind. At the same time, machine learning algorithms also have a significant impact on the research of the traditional computer security field [2]. In addition to using machine learning (ML) to build various malicious detections and attack identification systems, hackers may also use ML to achieve more accurate attacks. Recent studies have shown that many application fields, from computer vision to network security, are vulnerable to adversarial attack threats [3, 4].

Szegedy et al. [5] first proposed the concept of adversarial sample—an interesting weakness in neural networks. This paper has sparked widespread interest in adversarial attacks among researchers, and the number of adversarial attacks will continue to increase in the future as the economic benefits trend. Based on gradient descent and L-norm optimization methods, Liu et al. [6] first presented the method to exploit a malware-based visual detector to adversarial example attacks. Zhou et al. [7] were the first to propose an alternative model for training adversarial attacks without real data.

In order to provide a comprehensive reference tool for researchers to study adversarial attack, this paper classifies and compares adversarial attack algorithms of image, text, and malware fields. Moreover, the concepts, types, harms, evolvement trends of adversarial attack, and the commonly used defense technologies in recent years are reviewed systematically, and the future research direction is discussed [8, 9]. This paper is aimed at providing theoretical and methodological support for the research on adversarial attack security by extensively investigating the research literature on adversarial attacks and robustness [10, 11].

To sum up, the main contributions are as follows:

(1) Present adversarial attacks according to the idea of "Why? → What? → How?" In this way, researchers can quickly and efficiently establish the awareness of adversarial attack

(2) Portray a roadmap for carrying out adversarial attack research, which can help researchers quickly and efficiently access the domain of adversarial attacks security research

(3) In order to ensure the timeliness of the review, the most up-to-date and comprehensive references are provided based on the articles published in authoritative journals and top academic conferences after 2010

(4) In order to enable researchers to find referable methods as needed, the technical core and deficiency of methods in each typical method are introduced

(5) In order to quickly find the entry point of adversarial attack research, the pieces of literature on adversarial attack are classified from different perspectives.

The structure of the article is as follows. In "Why Study Adversarial Attack," we first recommend why we should study adversarial attacks. In "Adversarial Attack," we describe the concepts, classifications, hazards, and methods associated with adversarial attacks. A separate section is devoted to categorizing and comparing the adversarial attack methods of images, text, and malware. In "Defense Methods," we discuss defense methods. Then, in "Additional Complements," the additional supplements needed for adversarial attack research are presented. After that, we put forward a broader prospect of research direction in "Discussion." In "Comparison with Other Similar Reviews," the differences between this paper and other similar reviews are discussed. Finally, we concluded in "Conclusions."

2. Why Study Adversarial Attack

Choosing directions is the first step in conducting research. The research direction of adversarial attack security is reflected in the following aspects:

(1) Adversarial attack has become a serious threat to the current AI system. In the era of the Internet of Everything, the network has become the ideal goal for cyber attackers. Vulnerability in artificial intelligence is often exploited by attackers to launch cyberattacks. In 2018, a fake video of Obama railing against Trump went viral across the United States. Some criminals fabricate information to manipulate the election. With the frequent occurrence of these fraud incidents, governments and organizations of various countries have formulated and improved the relevant laws and regulations. In the United States, for example, the Deepfake Report Act was introduced in June 2019 and passed unanimously in the Senate as a separate bill only 120 days later. Despite the relentless efforts of the machine learning and AI communities to erect protective fences, the number of adversarial attacks is climbing significantly, and the threat posed by them continues to increase. Szegedy et al. [5] first proposed the concept of adversarial samples—intriguing weaknesses of neural networks. Their essay has sparked broad interest among researchers in adversarial attacks. Moreover, the number of adversarial attacks will continue to increase in the future, as economic interests evolve. Only by continuously strengthening the protective barrier of the deep learning model can a secure network security environment be built [12]

(2) The race of AI can promote adversarial attack security research. Attacking and defending adversarial machine learning is a process of iterative evolution. Adversarial attack creators have been probing new vulnerabilities, depicting new algorithms, and seeking new threats, while the defenders have been analyzing the features of new adversarial attacks and employing new methods to ensure efficient and effective defense against adversarial attacks. Consequently, choosing adversarial attack security as the research direction can not only merely keep the study at the forefront of AI security but also enable researchers to stimulate continuous motivation in the process of research.

3. Adversarial Attack

3.1. Concepts. In this subsection, some common terms used in the literature related to adversarial attacks are presented.

3.1.1. Adversarial Example. Adversarial example is an artificially constructed example that makes machine learning models misjudge by adding subtle perturbations to the original example but at the same time does not make human eyes misjudge [13].

3.1.2. White-Box Attack. White-box attacks assume that the target model can fully obtain the structure of the model, including the composition of the model and the parameters of the partition layer, and can fully control the input of the model [14].

3.1.3. Black-Box Attack. Black-box attacks have no idea of the internal structure of the model and can only control the input and carry out the next attack by comparing the feedback of the input and output [15].

3.1.4. Real-World Attack/Physical Attack. Real-world attacks/physical attacks do not understand the structure of the model and even have weak control over the input [16].

3.1.5. Targeted Attack. Targeted attacks will set the target before the attack, causing it to incorrectly predict the specific label of the adversarial images, which means that the effects after the attacks are determined [17].

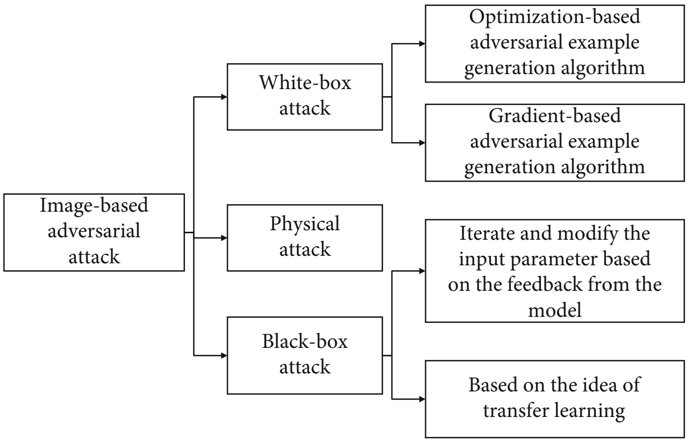
Image-based adversarial attack
White-box attack
Physical attack
Black-box attack
Optimization-based adversarial example generation algorithm
Gradient-based adversarial example generation algorithm
Iterate and modify the input parameter based on the feedback from the model
Based on the idea of transfer learning

Figure 1: Classification of image-based adversarial attack.

3.1.6. Untargeted Attack. Untargeted attacks do not need to set the target before the attack, as long as the result of identification is wrong after the attack [18].

3.1.7. Evade Attack. Evade attacks refer to adding disturbance to test samples and modifying the input during the test phase, to avoid or deceive the detection of the model and make the AI model unable to be correctly identified [19].

3.1.8. Poisoning Attack. By adding carefully constructed malicious examples in the model training phase, poisoning attacks make the trained model have backdoor or vulnerability, which can be used by attackers to carry out attacks [20].

3.1.9. Backdoor Attack. Backdoor attack means that the attacker can enter the system without identity authentication, which means that the attacker can bypass the security control to gain access to the system and even further damage the computer or network. The neural network backdoor attack referred to in this paper refers to that the attacker generates a model with a backdoor by implanting specific neurons in the neural network model, so that the judgment of the model on normal inputs is consistent with the original model, but the judgment on special inputs will be controlled by the attacker [21]. A backdoor attack is a type of poisoning attack [22].

3.1.10. Detection Model. The detection model means to detect whether the examples to be judged are adversarial examples by detecting components. Various detection models may judge whether the input is an adversarial example according to different criteria [23].

3.2. Classifications. In this section, we describe the main methods for generating adversarial examples in the image, text, and malware domains. And the work of some researchers is reviewed. Adversarial machine learning is a widely used technology in the field of image, and the research has been quite comprehensive. The malicious code domain and the text domain are similar and can be borrowed from each other. Figure 1 is the classification of image-based adversarial attack. According to the attacker's knowledge, the attack can be divided into black-box attack, white-box attack, and physical attack. Figure 2 is the classification of text-based adversarial attack. According to the access permissions of the model, the attack can be divided into black-box attack and white-box attack. And according to the effect after the attack, it can be divided into the targeted attack and non-targeted attack. Moreover, according to the text granularity, it can be divided into character level, word level, and sentence level. Besides, according to the attack strategy, it can be divided into image-to-text, importance-based, optimization-based, and neural network-based. Figure 3 is the classification of malware-based adversarial attack. According to the object, it can be divided into attacks on training data and attacks on neural network model.

3.3. Hazards. The main harm of adversarial attack includes the following:

(1) Model losing or stealing: network information data has become the most precious intangible asset in the current Internet of Everything era. In the current era of the Internet of Everything, network information data has become the most precious intangible asset. Plenty of adversarial attacks are designed to steal secret data, such as stealing privacy data from computers or servers and after that deceiving or blackmailing victims. More malicious attacks are aimed at enterprises to steal valuable trade secrets from enterprises and obtain economic benefits, even worse, targeting a country and stealing national security-related intelligence information from government departments for strategic purposes

(2) Model failure: the adversarial attack causes the failure of the deep learning model and makes it unable to work properly utilizing physical attack against the vulnerability of the model. For instance, in the field of autonomous driving, the superposition of image data with subtle perturbations makes it difficult for humans to recognize by senses, which leads to the wrong classification decision made by the machine learning model and causes traffic accidents

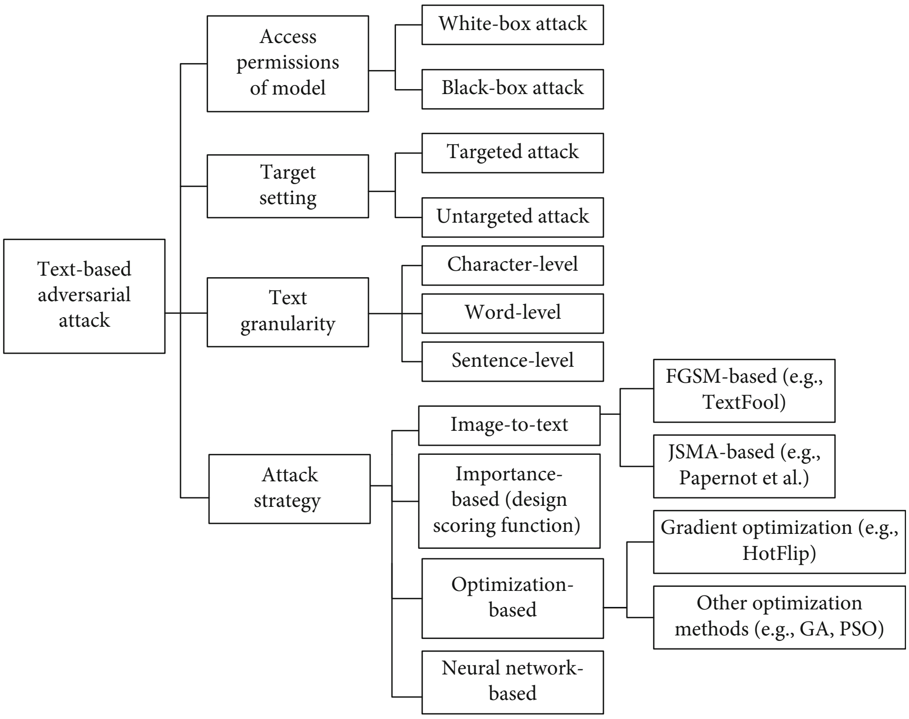
Text-based adversarial attack
Access permissions of model
White-box attack
Black-box attack
Target setting
Targeted attack
Untargeted attack
Text granularity
Character-level
Word-level
Sentence-level
Attack strategy
Image-to-text
FGSM-based (e.g., TextFool)
JSMA-based (e.g., Papernot et al.)
Importance-based (design scoring function)
Optimization-based
Gradient optimization (e.g., HotFlip)
Other optimization methods (e.g., GA, PSO)
Neural network-based

Figure 2: Classification of text-based adversarial attack.

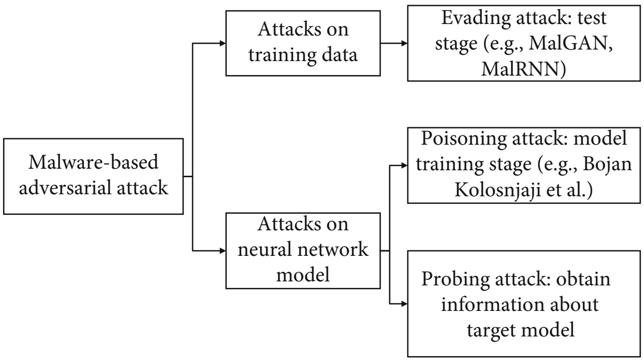
Malware-based adversarial attack
Attacks on training data
Evading attack: test stage (e.g., MalGAN, MalRNN)
Attacks on neural network model
Poisoning attack: model training stage (e.g., Bojan Kolosnjaji et al.)
Probing attack: obtain information about target model

Figure 3: Classification of malware-based adversarial attack.

(3) Data poisoning: a typical scenario is the recommendation system. When hackers add abnormal data to the training samples of deep learning, the most direct impact is that the classification error of the deep learning model, i.e., the recommendation system, generates an error

(4) Other hidden hazards: apart from the obvious harms mentioned above, adversarial attacks can also cause some hidden hazards. For instance, many malwares generate adversarial examples through some algorithms, and the malicious behavior does not change, but the detection of antivirus system fails, which eventually leads to malicious attacks on computers and networks as well as losses of profits.

3.4. Methods. The typical attack algorithms of each application domain are shown in Table 1.

3.4.1. Image-Based Adversarial Attack. Adversarial attacks are a major threat to computer vision. CCS 2020 has a total of four papers on adversarial machine learning, all of which are based on image research, three of which are about robustness and defense mechanisms, and only one is about adversarial attack methods. I have compared and summarized several papers of CCS, NIPS, USENIX, ECML PKDD, and JNCA from 2016 to 2020, as shown in Table 2.

Through reading the papers, it can be found that most of the current research in the field of image are optimized and improved based on previous research, and the attack types are generally divided into white-box attack, black-box attack, and physical attack [12].

3.4.2. Text-Based Adversarial Attack. Studies have shown that DNN models are susceptible to adversarial samples that lead to false predictions by adding imperceptible per to normal input. The study of adversarial samples in the field of images is abundant, but not enough in the field of text. Table 3 summarizes some of the relevant papers on the research of adversarial machine learning in the text field in recent years.

By reading the literature, the text domain adversarial samples can be divided into character level, word level, and sentence level. Attack types can be classified into white-box attack and black-box attack according to the model access rights. According to the effect after the attack, it can be divided into the targeted attack and untargeted attack. Attack strategies in the text domain generally include reference algorithms in the field of image, transformed into optimization problems, using gradient or designing scoring functions, and training neural network models to automatically generate adversarial samples [13].

Table 1: Typical attack algorithms for different application domains.

Method	Access permission	Targeted/nontargeted	Application domain	Metrics/strategies
Papernot et al. 2016 [24]	White box	Nontargeted	Text	Gradient
TextFool 2018 [25]	White and black box	Targeted	Text	Gradient
HotFlip 2018 [26]	White box	Nontargeted	Text	Gradient
Alzantot et al. 2018 [27]	Black box	Targeted	Text	Euclidean distance
DeepWordBug 2018 [28]	Black box	Nontargeted	Text	Scoring function
Zhao et al. 2018 [29]	Black box	Nontargeted	Text and image	WGAN-based
TextBugger 2019 [30]	White and black box	Nontargeted	Text	Confidence coefficient, scoring function
DISTFLIP 2019 [31]	Black box	Nontargeted	Text	Gradient
UPSET 2011 [32]	Black box	Targeted	Universal	l_∞
L-BFGS 2014 [33]	White and black box	Targeted	Image	l_∞
FGSM-based 2015 [34]	White box	Targeted/nontargeted	Universal	l_2,l_∞
JSMA 2015 [35]	White box	Targeted	Image	l_∞
DeepFool 2016 [36]	White box	Nontargeted	Image	l_2,l_∞
BIM and ILCM 2017 [37]	White box	Nontargeted	Image	l_0
One-pixel 2017 [38]	Black box	Targeted	Image	l_0
C&W 2017 [39]	White box	Nontargeted	Image	l_0,l_2,l_∞
Universal perturbations 2017 [40]	White box	Nontargeted	Universal	l_2,l_∞
ANGRI 2017 [41]	Black box	Targeted	Image	l_∞
Houdini 2017 [42]	Black box	Targeted	Image	l_2,l_∞
ATNs 2017 [43]	White box	Targeted	Image	l_∞
MalGAN 2017 [44]	Black box	Targeted	Malware	GAN-based, gradient
SLEIPNIR 2018 [45]	White box	Targeted	Malware	Saddle-point optimization
Kolosnjaji et al. 2018 [46]	White box	Targeted	Malware	Gradient
Song et al. 2020 [47]	Black box	Targeted	Malware	Number of bytes changed
Rosenberg et al. 2020 [48]	Black box	Targeted	Malware	API call-based, GAN
MalRNN 2020 [49]	Black box	Targeted	Malware	Varying the append size

3.4.3. Malware-Based Adversarial Attack. In recent years, machine learning has been used to detect malware, and malware developers have a strong incentive to attack the detection model. Table 4 summarizes some of the papers related to adversarial attacks in the field of malicious code in recent years.

Through literature reading, it can be found that attack types are generally divided into attacks on training data and attacks on neural network models. According to the attack scenario, it can be divided into white-box attack, gray-box attack, and black-box attack. In addition, the methods adopted include GAN-based, gradient-based, and heuristic-based algorithms [14].

4. Defense Methods

At present, relevant studies have been conducted to explore the security of AI itself, to ensure the integrity and confidentiality of AI models and data, so that it will not be easily affected by attackers to change judgment results or leak data in different scenarios. Hendrycks and Gimpel [64] proposed three methods for detecting adversarial images. Zhang et al. [65] proposed a hardware-assisted randomization method against adversarial examples to defend against various adversarial attacks. At present, AI attacks are divided into the evasive attack, poisoning attacks backdoor attack, and model-stealing attack. The backdoor attack is a kind of poisoning attack. Table 5 lists the various defense techniques of the AI system during data collection, model training, and model use phases. Furthermore, we need to continue to study AI interpretability, enhance our understanding of how ML works, and build institutional defense measures to build AI security platforms [66].

4.1. Defense Methods of Evasive Attack

4.1.1. Adversarial Training. Adversarial training is an important way to enhance the robustness of neural network models. The basic principle of this technique is to use known attack techniques to generate adversarial samples in the model training stage. Then, the adversarial samples are added to the training set of the model, and the model is iteratively retrained until a new model that can resist disturbance is generated. At the same time, since the synthesis of multiple types of adversarial samples increases the data of the training set, this technique can not only enhance the robustness of the

Table 2: Image-based adversarial attack.

Author	Solution	Cores	Shortcomings
Sharif et al. 2016 [50]	Mahmood Sharif et al. propose a class of attack that allows an attacker to avoid identifying or impersonating another individual. In addition, they described a method of automatically generating attacks and achieved it by printing a pair of glasses frames.	(1) Three DNNs were used (2) Use gradient descent algorithm to optimize and find appropriate disturbance (3) Facilitate physical realizability by using facial accessories and adjusting the mathematical formula of the attacker's target.	Their attacks need to be improved in the face of black-box face recognition systems and the most advanced face detection systems.
Nicolas Papernot et al. 2017 [51]	Nicolas Papernot et al. propose a practical black-box attack based on a new substitute training algorithm which is using synthetic data generation to produce adversarial examples misclassified by black-box DNNs.	(1) Obtain the training set of the substitute detector (2) Select the appropriate substitute detector model structure (3) Iteratively train the substitute detector (4) Attack the substitute detector to generate adversarial example.	Adversarial training can effectively defend against this black-box attack algorithm.
Shafahi et al. 2018 [52]	Ali Shafahi et al. introduce "clean-label" poisoning attacks, which do not require attackers to have any control over the labeling of training data. Moreover, in order to optimize the poisoning attack, a "watermark" strategy is proposed.	(1) Crafting poisoning data through feature conflict (2) The optimization algorithm uses the forward-backward-splitting iterative procedure (3) Add a low opaque watermark of the target instance to the poisoning instance to enhance the effect of poisoning attack.	The attack method they proposed will cause the unchanged target instance to be misclassified as a basic, and the side effects of adversarial training are worthy of further study.
Mirsky et al. 2019 [53]	Yisroel Mirsky et al. construct a framework (CT-GAN) based on deep learning. In their strategies, attackers can use the framework to automatically tamper with 3D medical images, injecting/removing lung cancer into/from CT scans.	(1) Capture data using attack vectors (2) Select the location of injecting/removing cancer (3) Use 3D spline interpolation to scale (4) Equalization and standardization are achieved by means of histogram and formula (5) Create samples → reverse preprocessing → add Gaussian noise → gain the complete slice → repeat steps/return data.	Medical scans are different from camera images, and further research on how to apply these techniques to detect attacks such as CT-GAN is needed.
Chen et al. 2019 [54]	Shang-Tse Chen et al. propose ShapeShifter, which uses physical perturbations to fool image-based target detectors like Faster R-CNN.	(1) In their strategies, by studying the Faster R-CNN algorithm, the nondifferentiability of the model was overcome, and gradient descent and backpropagation were successfully used to perform optimization-based attacks (2) ShapeShifter can generate adversarial perturbed stop signs, which are consistently misdetected by Faster R-CNN as other targets, posing a potential threat to computer vision systems.	A series of experiments show that their attacks fail to transfer, and further research is needed in the future.
Xiao et al. 2019 [55]	Qixue Xiao et al. propose an attack to automatically generate camouflage images against image-scaling algorithms. Both white-box and black-box scenarios can be applied.	(1) The surjective function is applied to generate the attack image from the target image (2) Automatic scaling attack: get coefficient matrix; find the perturbation matrix (3) Disturbance is obtained through concave-convex optimization.	Several defense strategies need further research and implementation.

Table 2: Continued.

Author	Solution	Cores	Shortcomings
Wang et al. 2020 [56]	Yajie Wang et al. describe a black-box attack method based on DNN object detection models, which is called evaporate attack. Moreover, experimental results show that their approach is superior to boundary attack on both 1-stage and 2-stage detectors.	(1) In their research, the GA-PSO algorithm is designed to resolve the issue of attacking black-box object detector with only position and label information (2) Add pixel optimal position guidance and random Gaussian noise to the velocity iteration formula.	If the model owner performs some processing on the output of the model (such as only provides the label of the object), the attack could be affected.
Solano et al. 2020 [57]	Jesús Solano et al. propose an intuitive attack method for mouse-based behavioral biometrics and compare it to black-box adversarial attack.	(1) Feature engineering: angle feature, dynamic feature (2) Authentication system: a set of binary classification model is designed to recognize a specific user's MBB (mouse-based behavioral biometric recognition) (3) Attacks: provides SCRAP and adversarial machine learning black-box attack.	An automated procedure for inverse feature calculation is needed to make the effectiveness of the comparative adversarial method more accurate.

Table 3: Text-based adversarial attack.

Author	Solution	Cores	Shortcomings
Ebrahimi et al. 2018 [26]	Javid Ebrahimi et al. propose a method of generating white-box adversarial examples, HotFlip, to make the character-level classifier classification errors.	(1) Use beam search to find a set of operations (flip, insert, and delete) that obfuscate the classifier (2) Calculate the directional derivative of the operation using the gradient represented by the one-hot input vector to estimate the change in the loss.	Failure to evaluate the robustness of different character-level models for different tasks The context is not well considered.
Gao et al. 2018 [28]	Ji Gao et al. introduce a new algorithm for generating text perturbed in black-box scenarios: DeepWordBug, which causes the deep learning classifier to misclassify text input.	(1) Use the scoring function to determine the importance of each word to the classification results and rank the words according to their ratings (2) Use the transformation algorithm to change the words selected.	This paper does not discuss the application of the algorithm in the white-box scenario.
Cheng et al. 2018 [58]	In this paper, the Seq2Sick framework is proposed to generate adversarial examples for sequence-to-sequence (seq2seq) model. Nonoverlapping attack and targeted keywords attack are mainly studied.	(1) A projection gradient method is proposed to solve the discrete problem of input space (2) Adopting group lasso to enhance the sparsity of distortion (3) Developed a regularization technique to improve the success rate.	The success rate of targeted one-keyword attack is reduced when the model of subword transformation is attacked.
Li et al. 2019 [30]	This paper proposes a general attack framework for generating adversarial text: TextBugger.	(1) White box: find the most important word through the Jacobian matrix, generate five types of bugs, and find the best one based on confidence (2) Black box: find the most important sentences first and then use the scoring function to find the most important words (3) Assessment: sentiment analysis and harmful content detection.	(1) This paper only performs nontarget attack and does not involve target attack (2) The integration of defense systems based on language perception or structure perception can be further explored to improve robustness.
Hang et al. 2019 [59]	In this paper, Metropolis-Hastings sampling (MHA) is proposed to generate adversarial samples for natural language.	(1) Black-MHA: select words by traversing the index for conversion operation and select the most likely words according to the score (2) White-MHA: the difference between the white-box attack and the black-box attack is preselection, which introduces gradients into the score calculation.	(1) It may produce incomplete sentences (2) Unrestricted entity and verb substitution also have a negative impact on the adversarial example generation of tasks (such as NLI).
Zang et al. 2020 [60]	In this paper, a new black-box adversarial attack model is proposed to solve the combinatorial optimization problem of word-level adversarial attack.	(1) A word substitution method based on the minimum semantic unit sememe is designed to reduce the search space (2) A search algorithm based on particle swarm optimization is proposed to search adversarial examples.	The improvement of robustness and the use of sememe in defense model need further study.

Table 4: Malware-based adversarial attack.

Author	Solution	Cores	Shortcomings
Hu et al. 2017 [44]	This paper proposes a GAN-based malware adversarial example generation algorithm (MalGAN), which can bypass the detection model based on black-box machine learning.	(1) Sampling malicious examples (2) Generating adversarial examples (3) Sampling benign examples (4) Labeling (5) Updating the weights according to the gradient.	This paper does not discuss the application of the algorithm in the white-box scenario.
Raff et al. 2017 [61]	Edward Raff et al. developed the first network architecture that could successfully process over 2 million steps of raw byte sequences.	Architecture features: (1) Expand with sequence length (2) The ability to consider local and global environments when examining the entire file (3) Helps to analyze the interpretive ability of flagged malware.	The standardization of batch processing needs to be further explored.
Al-Dujaili et al. 2018 [45]	This paper proposes the SLEIPNIR framework, which uses saddle-point optimization to learn the malware detection model of executable files represented by binary encoding features.	(1) Frame construction based on saddle-point optimization (2) Add randomization to the method (3) An on-line measurement method is introduced.	No instructions were given on how to locate benign samples.
Kolosnjaji et al. 2018 [46]	This paper proposes a gradient-based evading attack.	(1) Adds a set of bytes to the end of the binary file to generate adversarial examples that do not break the malicious functionality of the source file (2) Initializes the iteration counter, repeatedly sets the number of filled bytes and calculates the gradient.	The dataset is not large enough. The grain size is not fine.
Song et al. 2020 [47]	This paper presents a systematic framework for creating and evaluating real malware in order to achieve evasive attack.	(1) Adversarial example generation: design action set and verification function (2) Minimize action sequence (3) Feature interpretation.	The defend methods and robustness of the framework are less discussed.
Rosenberg et al. 2020 [48]	In this paper, a black-box attack against API-based machine learning malware classifiers is proposed.	(1) Use valid parameters with no operation effect (2) Determine the increase in the number of API calls using the method of logarithmic transformation backtracking (3) Use GAN to select generated API calls (4) Use the adaptive evolutionary algorithm to realize the attack with high query efficiency based on the score.	Defense mechanisms are not discussed.
Ebrahimi et al. 2020 [49]	This paper presents MalRNN, a novel deep learning-based approach to automatically generate evasive malware variants.	(1) Obtain data through system sampling (2) Learn the language model from benign malware binaries using character-level sequence-to-sequence RNN (3) Ensure the ability to generate malware variants	(1) There is no discussion of defense mechanisms (2) The antivirus avoidance method is simple
Nguyen et al. 2020 [22]	Thien Duc Nguyen et al. demonstrate that federated learning-based IoT intrusion detection systems are vulnerable to backdoor attacks and proposed a new kind of data poisoning attack.	By injecting a small amount of malicious data into the training process using only the compromised IoT device (rather than the gateway/client) and remaining undetected, the model is gradually backdoor.	Existing defense methods are ineffective against this attack, so new defense mechanisms are needed to defend against it.

Table 4: Continued.

Author	Solution	Cores	Shortcomings
Demetrio et al. 2020 [62]	Luca Demetrio et al. propose a general framework called RAMEn for performing black and white-box adversarial attacks on Windows malware detectors based on static code analysis.	(1) Two new attacks—Extend and Shift—were proposed to extend the DOS header and transfer the contents of the first part, respectively, according to the adversarial load of the injection (2) The experimental results show that the proposed attack improves the tradeoff between the probability of avoidance and the number of bytes manipulated in the white-box and black-box attack settings.	Attackers cannot arbitrarily add adversarial loads because proposed content injection attack must adhere to certain restrictions imposed by the format.
Chen et al. 2020 [63]	This paper presents Android HIV, an automated tool for creating adversarial examples on the Android Malware Detector based on machine learning.	(1) Attack on MAMADROID: optimize the target function and modify the C&W algorithm; Jacobian matrix is calculated and JSMA algorithm is refined (2) Attack on DREBIN: generate adversarial examples based on Jacobin.	There is no in-depth analysis of defense mechanisms against such attacks. Nor has the effectiveness of the different alternative model architectures been compared.

Table 5: AI security defense technology.

Type	Phase Data collection phase	 Model train phase	 Model usage phase
Evasive attack	Generating adversarial examples	Network distillation; adversarial training	Adversarial examples detection; input reconstruction; DNN model validation
Poisoning attack	Filtering training data; regression analysis	Integration analysis	
Back door attack		Model pruning	Input preprocessing
Model-stealing attack	Differential privacy	Privacy aggregation teacher model; model watermarking	

newly generated model but also enhance the accuracy and standardization of the model [67].

Ju et al. [68] propose E-ABS (analysis-by-synthesis) which can extend the ABS robust classification model to more complex image domains. The core contents include the following: (1) generation model: Adversarial Auto Encoder (AAE) is used to evaluate the class-conditional probability; (2) discriminative loss: use the discriminative loss during training to expose the conditional generation model to unevenly distributed samples; (3) variational inference: the ABS-like model estimates the likelihood of each category by maximizing the likelihood estimation; and (4) lower bound for the robustness of E-ABS: the lower bound of the nearest adversarial example to E-ABS is derived by using the ABS model. Nevertheless, the generation model is sensitive to image similarity measurements. And the reasoning and operation efficiency of ABS-like model on large datasets are low.

Chen et al. [69] are the first to evaluate and train the verifiable robust properties of PDF malware classifiers. They proposed a new distance metric in the PDF tree structure to construe robustness, that is, the number of different subtrees of depth 1 in two PDF trees. Furthermore, their experimental results show that the most advanced and new adaptive evolutionary attackers need 10 times the L0 feature distance and 21 times the PDF operation to evade the robustness model. However, the tradeoff between multiple robustness attributes and training costs needs further study.

4.1.2. Network Distillation. Distillation is a method of compacting the knowledge of a large network into smaller networks. Specialist models means, for a large network, multiple specialized networks can be trained to improve the model performance of the large network. The practice of distillation is generally to train a large model (teacher network) first and then heat the large model. The output of the large model is used as a soft target, and the real label of data is used as a hard target. The two are combined to train the small model (student network) [70].

The basic principle of the network distillation technique is to series multiple DNNs in the model training stage, in which the classification results generated by the former DNN are used to train the latter DNN. Papernot et al. [71] found that the transfer knowledge could reduce the sensitivity of the model to subtle disturbances to some extent and improve the robustness of the AI model. Therefore, network distillation technology is proposed to defend against evasive attacks, and it is tested on MNIST and CIFAR-10 datasets. It is found that network distillation technology can reduce the success rate of specific attacks (such as JSMA and FGSM).

4.1.3. Adversarial Example Detection. The principle of adversarial example detection is to detect whether the example to be judged is an adversarial example by adding the detection component of the external detection model or the original model in the usage stage of the model. Before the input example reaches the original model, the detection model will determine whether it is an adversarial example. The detection model can also extract relevant information from each layer of the original model and synthesize various information to carry out detection. Various detection models may use different criteria to determine whether the input is an adversarial example [72].

Shumailov et al. [73] propose a new provable adversarial example detection protocol, the Certifiable Taboo Trap (CTT). They extended the Taboo Trap method. Moreover, three different CTT modes (CTT-lite, CTT-loose, and CTT-strict) are also discussed. However, this scheme cannot be used to defend against a specific adversarial example, thus resulting in a more flexible and universal defense mechanism.

4.1.4. Input Reconstruction. The principle of input reconstruction is that the input samples are transformed to resist evasive attack in the use stage of the model, and the transformed data will not affect the normal classification function of the model. Reconstruction methods include noising, denoising, preprocessing, gradient masking, and using autoencoder to change the input examples [74]. At the same time, I think the metamorphosis of the sample has some similarities with the evolution of malicious code. Interestingly, 8 papers presented at the ICLR 2018 conference used gradient masking, but researchers quickly cracked 7 of them.

4.1.5. DNN Verification. Similar to software verification analysis technology, DNN verification technology uses solvers to verify various properties of DNN models, such as verifying that there is no adversarial example within a specific perturbation range. Nevertheless, the DNN model is usually verified as an NP-complete problem, and the efficiency of solver is low. Through selection and optimization, such as priority selection of model node validation, sharing of validation information, and validation by region, the operation efficiency of DNN verification can be further improved [75].

4.1.6. Data Augmentation. In reality, we often encounter the situation of insufficient data. The essence of data augmentation is to expand the original training set with generated adversarial samples when massive data is lacking, so as to ensure effective training of the model [76].

4.2. Defense Methods of Poisoning Attack

4.2.1. Training Data Filtering. This technique focuses on the control of the training dataset and uses detection and purification methods to prevent the poisoning attack from affecting the model. Specific directions include the following [77]: finding possible poisoning attack data points according to the label characteristics of the data and filtering these attack points during retraining; the model contrast filtering method was used to reduce the sampling data that could be used by poisoning attack, and the filtering data was used to against poisoning attack.

4.2.2. Regression Analysis. This technique is based on statistical methods that detect noise and outliers in datasets. Specific methods include defining different loss functions for the model to check outliers, using the distribution characteristics of data for detection [78], etc.

4.2.3. Ensemble Learning. Ensemble learning is to build and combine multiple machine learning classifiers to improve the ability of the machine learning system to resist poisoning attacks. Multiple independent models jointly constitute the AI system. And the possibility of the whole system being affected by the poisoning attack is further reduced due to the different training datasets adopted by multiple models [79].

Liao et al. [80] design an adaptive attack method to study the effectiveness of integrated defense based on transformation for image classification and its reasons. They propose two adaptive attacks to evaluate the integrated robustness of reversible transformation: TAA (adaptive attack based on transferability) and PAA (perturbation aggregation attack). Moreover, the ensemble evaluation method is used to evaluate the ensemble robustness of the irreversible transformation. However, the experimental results show that the integrated defense based on transformation is not enough to resist the antagonistic samples, the defense method is not reliable, and further efforts are needed.

4.2.4. Iterative Retraining. Iterative retraining refers to the iterative training of neural networks. Adversarial examples are generated according to any attack model and added to training data. Then, the neural network model is attacked, and the process is repeated [81].

4.3. Defense Methods of Back Door Attack

4.3.1. Input Preprocessing. The purpose of this method is to filter the input that can trigger the back door and reduce the risk of the input triggering the back door and changing the model judgment [82]. Data can be divided into discrete and continuous types. The image data is continuous and easy to encode as a numerical vector. The pretreatment operation is linear and differentiable, and there are many operation methods, such as mean standardization. Text data is discrete and symbolized. The preprocessing operation is nonlinear and nondifferentiable. One-hot is generally used for pretreatment.

4.3.2. Model Pruning. Pruning in neural networks is inspired by synaptic pruning in the human brain. Synaptic pruning, complete decline and death of axons and dendrites, is a synaptic elimination process that occurs between childhood and the onset of puberty in many mammals, including humans. The principle of model pruning is to cut off the neurons of the original model appropriately, reducing the possibility of the backdoor neurons working under the condition that normal function is consistent. Using a fine-grained pruning method [83], the neurons that make up the backdoor can be removed, and the backdoor attack can be prevented.

4.4. Defense Methods of Model-Stealing Attack

4.4.1. PATE. The basic principle of PATE is to divide the training data into multiple datasets without intersection in the model training stage, and each set is used to train an independent DNN model (called teacher model). These independent DNN models are then used to vote together to train a student model [84]. This technology ensures that the judgment of the student model will not disclose the information of a particular training data, so as to ensure the privacy of the training data.

4.4.2. Differential Privacy. In the model training stage, the method is used to add noise to the data or the model training step by conforming to the differential privacy [85]. For example, Giraldo et al. [86] are the first to solve the adversarial classification problem in a system that uses differential privacy to protect the user's privacy. They find an optimal fake data injection attack that reduces the system's ability to detect anomalies, while allowing the attacker to remain undetected by "hiding" the fake data in the differential privacy noise. Besides, they design the optimal defense method to minimize the impact of such attack. They also show how the optimal DP-BDD (differential privacy–bad-data detection) algorithm achieves a Nash equilibrium between attackers trying to find the optimal attack distribution and defenders trying to design the optimal bad-data detection algorithm. Yet, the trade-offs between security, privacy, and practicality need to be further explored. Differential privacy adds noise to sensitive data or calculations performed on sensitive data to ensure privacy without unduly reducing the utility of the data.

4.4.3. Model Watermarking. The technique is to embed special labels in the original model during the model training stage. If a similar model is found, a special input sample can be used to identify whether the similar model was obtained by stealing the original model. Adi et al. [87] proposed a black-box deep neural network watermarking method. The robustness of the watermarking algorithm is evaluated under black-box and gray-box attacks.

In addition to the defensive measures mentioned above, there are other ways to improve the model's robustness against attacks, such as data compression, data randomization, data secondary sampling, outliers removal, model

Table 6: Common datasets for image adversarial attack.

Type of dataset	Data source	Application instances
Publicly accessible dataset	ImageNet	Xiao et al. 2019 [55]; Ma et al. 2019 [92]
	MNIST	Demontis et al. 2019 [20]; Ling et al. 2019 [93]; Fang et al. 2020 [94]; Ma et al. 2019 [92]; Moosavi-Dezfooli et al. 2016 [36]; Yang et al. 2019 [95]
	CIFAR-10	Ling et al. 2019 [93]; Shafahi et al. 2018 [52]; Ma et al. 2019 [92]; Moosavi-Dezfooli et al. 2016 [36]; Yang et al. 2019 [95]
	CH-MNIST; Fashion-MNIST; Breast Cancer Wisconsin	Fang et al. 2020 [94]
	VidTIMIT database	Korshunov et al. 2018 [96]
	WebFace; VGGFace2	Shan et al. 2019 [97]
	FaceScrub	Yang et al.2019 [95]; Shan et al. 2019 [97]
	PubFig	Sharif et al. 2016 [50]; Shan et al. 2019 [97]
	Cora; Citeseer; Polblogs	Jin et al. 2020 [98]
	Social Face Classification (SFC) dataset	Taigman et al. 2014 [99]
	MS-COCO	Chen et al. 2019 [54]
	CelebA	Yang et al. 2019 [95]
	MS-COCO 2017; PASCAL VOC 2007; PASCAL VOC 2012	Wang et al. 2020 [56]
	Labeled Faces in the Wild (LFW) database	Demontis et al. 2019 [20]; Taigman et al. 2014 [99]; Ma et al. 2019 [92]
	YouTube Faces (YTF) dataset	Taigman et al. 2014 [99]
	LIDC-IDRI dataset	Mirsky et al. 2019 [53]
	ILSVRC 2012	Simonyan et al.2015 [100]; Moosavi-Dezfooli et al. 2016 [36]
Commercial dataset	Fugazi	Din et al. 2018 [101]
Artificially generated dataset	Generated by toolkits manually	Yu et al. 2020 [102]

regularization, deep contractive network, biologically inspired conservation [32], attention mechanism [88], GAN-based, magnet [89], and high-level representation guided denoiser (HGD) [90].

Each of the above defense techniques has specific application scenarios and cannot completely defend against all the adversarial attacks. We can consider the above defense technology in parallel or serial integration to see whether the defense effect is better. For instance, data augmentation has the flexibility to easily plug in other defense mechanisms [91].

5. Additional Complements for Adversarial Machine Learning Research

Apart from focusing on the research methods of adversarial attack and defense, we also need to know other sides of this field.

5.1. The Choice of Dataset. By analyzing the already published articles, there are three types of datasets used in the adversarial machine learning research community currently. The common dataset for image adversarial machine learning research is shown in Table 6. The application of the text adversarial machine learning dataset is shown in Table 7. And Table 8 shows the application of the malware adversarial machine learning dataset.

5.1.1. Publicly Accessible Dataset. At present, the majority of published papers use publicly accessible datasets from the Internet. These datasets are free to use and are maintained and updated by researchers in the field of computer science.

5.1.2. Commercial Dataset. Commercial datasets are generally not freely utilized or publicly available.

5.1.3. Artificially Generated Dataset. Other datasets are manually generated or crawled from the website by researchers using special tools.

5.2. General Adversarial Machine Learning Tools. Some commonly used tools can be used to assist experimental verification in the experimental phase of adversarial machine learning research. Common tools for adversarial ML include sandboxes, Python-based tools, and Java-based tools.

5.2.1. Sandbox. The Sandbox is a virtual system application that allows the user to run a browser or other application in a Sandbox environment, so the changes that result from running it can be deleted later. It creates a separate operating environment that restricts program behavior according to

Table 7: Common datasets for text adversarial attack.

Type of dataset	Data source	Application instances
Publicly accessible dataset	AG's news; SST	Ebrahimi et al. 2018 [26]; Sato et al. 2018 [103]
	IMDB	Gao et al.2018 [28]; Zhang et al. 2020 [104]; Zang et al. 2020 [60]; Li et al. 2019 [105]; Neekhara et al. 2018 [104]
	SNLI	Zhang et al. 2020 [104]; Zang et al. 2020 [60]
	SST-2	Zang et al. 2020 [60]
	Enron spam emails	Gao et al. 2018 [28]
	Rotten Tomatoes Movie Reviews	Li et al. 2019 [105]
	DUC2003; DUC2004; IGAWORD	Cheng et al. 2020 [58]
	Sogou News; DBPedia; Yahoo! Answers; Amazon Review	Sato et al. 2018 [103]
Commercial dataset	Kaggle	Li et al. 2019 [105]

Table 8: Common datasets for malware adversarial attack.

Type of dataset	Data source	Application instances
Publicly accessible dataset	VirusShare; Citadel; APT1	Kolosnjaji et al. 2018 [46]; Al-Dujaili et al. 2018 [45]
	VirusTotal	Song et al. 2020 [47]; Huang et al. 2019 [106]; Suciu et al. 2019 [107]
	Drebin	Xu et al. 2020 [108]; Chen et al. 2020 [63]; Demontis et al. 2019 [20]; Arp et al. 2014 [109]
	https://malwr.com/	Hu et al. 2017 [44]
	NSL-KDD	Zhang et al. 2020 [110]
	MAMADROID	Chen et al. 2020 [63]
	EMBER	Suciu et al. 2019 [107]
	MasterDGA; Alexa site	Alaeiyan et al. 2019 [111]
Commercial dataset	The Kaggle Malware dataset of Microsoft	Salem et al. 2019 [112]; Yan et al. 2018 [113]
	McAfee Labs	Huang et al. 2019 [106]
	FireEye; Reversing Lab	Suciu et al. 2019 [107]
	Microsoft's antimalware team	Stokes et al. 2018 [114]
Artificially generated dataset	Generated by toolkits manually	Suciu et al. 2019 [107]

security policies, and programs that run inside of it do not permanently affect the hard disk. In cyber security, a sandbox is a tool used to handle untrusted files or applications in an isolated environment. For instance, Hu et al. used the Cuckoo Sandbox to process malware samples.

5.2.2. Machine Learning Tools Based on Python. Python is regarded as the best suited programming language for ML. Therefore, a series of Python-based machine learning and deep learning tools have been developed by researchers in the adversarial ML.

5.2.3. TensorFlow. TensorFlow is an end-to-end open source machine learning platform. It has a comprehensive and flexible ecosystem of tools, libraries, and community resources that will help researchers drive the development of advanced machine learning technologies. Besides, it enables developers to easily build and deploy applications powered by machine learning.

5.2.4. Keras. Keras is a high-level neural network API written in Python. It runs with TensorFlow, CNTK, or Theano as a backend. The focus of Keras development is to support rapid experimentation. Being able to turn ideas into results with minimal delay is the key to carry out research. With it, deep network can be built quickly and training parameters can be selected flexibly.

5.2.5. PyTorch. PyTorch is based on Torch and is used for applications such as natural language processing. It is a tensor library optimized using GPU and CPU. Moreover, it can be regarded as a powerful deep neural network with automatic derivation function.

5.2.6. NumPy. NumPy is a basic package for scientific computing using Python. It can be used to store and process large matrices and support a large number of dimension arrays and matrix operations. In addition, it also provides a large number of mathematical libraries for array operations.

Table 9: The connections between papers.

Type	Author	Time/published in	Connection
Image	Mahmood Sharif et al. [50]	2016/CCS	In the field of image, papers of CCF A and B conferences and top journals in the field of security were selected, covering the papers of nearly five years from 2016 to 2020, involving offense and defense, and being able to understand the frontier research in the field of adversarial machine learning.
	Nicolas Papernot et al. [51]	2017/CCS	
	Ali Shafahi et al. [52]	2018/NIPS	
	Yossi Adi et al. [87]	2018/USENIX	
	Yisroel Mirsky et al. [53]	2019/USENIX	
	Shang-Tse Chen et al. [54]	2019/ECML PKDD	
	Qixue Xiao et al. [55]	2019/USENIX	
	Yajie Wang et al. [56]	2020/JNCA	
	Jesús Solano et al. [57]	2020/CCS	
	Chang Liao et al. [80]	2020/CCS	
	Ilia Shumailov et al. [73]	2020/CCS	
	An Ju et al. [68]	2020/CCS	
Text	Javid Ebrahimi et al. [26]	2018/ACL	In the field of text, 6 papers including [30] are selected, with [30] as the core. Among the 40 related papers, 6 papers are selected, among which three [26, 58, 115] quoted by [30] are cited by [59, 60].
	Ji Gao et al. [115]	2018/SPW	
	Minhao Cheng et al. [58]	2018/AAAI	
	Jinfeng Li et al. [30]	2019/NDSS	
	Huangzhao Zhang et al. [59]	2019/ACL	
	Yuan Zang et al. [60]	2020/ACL	
Malware	Weiwei Hu et al. [44]	2017/arXiv	These papers are about malware adversarial machine learning found through connected papers. Among them, [44–46] are the papers based on windows platform. [48] is the paper related to binary malware adversarial examples. [47, 49] are similar to the same review on Windows PE file generation adversarial examples. The remaining papers include top conference papers, top journal papers, and arXiv in the field of security, among which [69] is about robustness research paper.
	Edward Raff et al. [61]	2017/arXiv	
	Abdullah Al-Dujaili et al. [45]	2018/arXiv	
	Bojan Kolosnjaji et al. [46]	2018/arXiv	
	Wei Song et al. [47]	2020/arXiv	
	Ishai Rosenberg et al. [48]	2020/ACSAC	
	Mohammadreza Ebrahimi et al. [49]	2020/AAAI	
	Thien Duc Nguyen et al. [22]	2020/DISS	
	Luca Demetrio et al. [62]	2020/arXiv	
	Xiao Chen et al. [63]	2020/TIFS	
	Yizheng Chen et al. [69]	2020/USENIX	

5.2.7. Scikit-Learn. Scikit-Learn is a machine learning tool based on Python with simplicity and efficiency. It can perform data mining and data analysis. Everyone can access it, and it is open source. It includes the following six basic functions: classification, regression, clustering, dimensionality reduction, model selection and preprocessing [115].

5.2.8. Machine Learning Tools Based on Java. Java-based machine learning platforms include WEKA, KNIME, and RapidMiner. WEKA provides Java's graphical user interface, command-line interface, and Java API interface. It is probably the most popular Java machine learning library. Apache OpenNLP is a toolkit for processing natural language text. It provides methods for natural language processing tasks such as tokenization, segmentation, and entity extraction.

5.3. The Connections between Papers. Table 9 shows the relationships among some of the literatures listed in this paper.

6. Discussion

The competition between offense and defense in the security field is endless. The previous sections provide an introduction to adversarial attacks and defenses, so that the reader can learn about them. Next, there is our discussion and outlook in this area.

6.1. Adversarial Example Generation Based on Malware. First, let us introduce the "feature engineering" of malicious code:

(1) Digital feature extraction: scale, normalize, and MinMaxScaler

(2) Text feature extraction: word set model and word bag model

(3) Data extraction: CSV is the most common format.

Adversarial machine learning is a widely used technique in the image domain. The adversarial attack technology in the field of the image is becoming more and more mature. In the future, we plan research adversarial attack samples in the field of malicious code. Since the structure of malicious code is similar to that of text data, we can consider transferring the text adversarial attack algorithm to the field of malicious code. There are two main ways to generate text adversarial samples. One is to generate adversarial samples directly through text editing operations such as insert, delete, and replace by using the characteristics of the text. The other is to map the text data into continuous data and generate the adversarial samples by using some algorithms in the field of computer vision for reference. Yan et al. [116] propose a genetic algorithm-based malicious code adversarial sample generation method to static rewrite PE files. The atomic rewrite operation screened by the fuzzy test is similar to the text edit operation.

6.2. Adversarial Example Generation Based on Swarm Intelligence Evolutionary Algorithm. Swarm intelligence evolutionary algorithm is a heuristic computing method that simulates the swarm behavior of insects, birds, and fish in the biological world, including genetic algorithm, ant colony algorithm, particle swarm algorithm, and cultural algorithm. Currently, Liu et al., Yan et al. [116], and Wang et al. [56] have all generated adversarial samples by improving swarm intelligence evolutionary algorithm. However, the literature review found that there are few articles about the cultural algorithm, among which no one has conducted the research of adversarial sample generation based on cultural algorithm. This is one of our future research directions.

6.3. Malware Evolution. Adversarial sample generation in the domain of malicious code is, in my opinion, similar to the evolution of malicious code. In order to counter the network security protection system, malicious code makers continue to use new technologies and new methods to create new malicious code. As a result, the malicious code is constantly evolving to ensure that it can evade security systems. Taking the evolutionary process of a family sample as an example, the sample population within a family can be regarded as a spatiotemporal set sequence, and the sample sets generated at different stages have different functional characteristics. Samples within each set will adopt different evolutionary methods to carry out internal upgrading, and different sets will also adopt collaboration methods to carry out coevolution. The goal of its evolution is to ensure its continued survival ability and attack ability to complete destructive tasks under different network security protection environments. The generated adversarial samples can be seen as a form of evolution of the malicious code. More attention can be paid to this research in the future.

6.4. Improve the Transportability. Transportability does not mean that a program can be written without modification to any computer, but rather that a program can be written without many modifications when conditions change. Transportability is one of the system quality assurances. It reflects the universality of the model. And transportability means that an adversarial sample generated by a neural network model on one dataset can also successfully attack another neural network model or dataset. Generally divided into three types:(1) the same architecture, different datasets (for example, both are based on Windows platform but use PE file and APK file, two types of datasets, respectively); (2) the same dataset with different architectures (for example, both are PE files but are applied to Windows and iOS architectures); and (3) different datasets and architectures. Although some models have successfully achieved portability, performance is still declining. Therefore, it is worth studying to improve the portability of the model.

6.5. Automation. Although some researchers have achieved automatic adversarial sample generation, many others craft adversarial samples manually. The artificial way is time-consuming and laborious and does not conform to the trend and new requirements of the development of The Times. In the future, with the efforts of researchers, I believe this problem will be greatly improved.

6.6. Possible Defensive Measures. In order to ensure system security in the field of AI, how to defend against attacks is the focus of current research. Many good researchers have designed powerful attacks but have not come up with effective countermeasures to defend them. The attack mentioned in this article must be carried out on the premise that the adversary can access the software or system. If the security of the access control is done well, it is helpful to protect the security of AI. In the stage of access control, identity authentication technology is one of the most important links. Secure multiparty computation (MPC) and homomorphic encryption (HE) are important privacy protection technologies in identity authentication systems [117–121]. I envision a combination of MPC and fully homomorphic encryption (FHE) to defend against attacks. MPC and FHE have high security. Zheng et al. [122] design and build Helen, a system that can achieve malicious security collaborative learning. They greatly reduced the model training time for achieving stochastic gradient descent (SGD) under MPC's SPDZ protocol by using the alternating direction multiplier method (ADMM) and singular value decomposition (SVD). Giraldo et al. [86] have solved the adversarial classification problem in a system that uses differential privacy to protect user privacy. Differential privacy is a noise-based MPC. Besides, the homomorphic encryption model is an effective way to improve data security and availability, which means the behaviors of users will not be leaked when trusted third parties help users process their data. To sum up, it is worth trying to combine secure multiparty computing and fully homomorphic encryption to defend against attacks. In addition, the blockchain technology has been mature and widely used in various fields, which can be regarded as a solution for examining malicious input and can be combined with MPC [123–125].

7. Comparison with Other Similar Reviews

As shown in Table 10, there are several similar surveys of the adversarial attack. The differences between this article and the other are as follows:

Table 10: Shortcomings of similar reviews.

Author	Main content	Shortcomings
Guofu Li et al. 2018 [126]	This article introduces the concepts and types of adversarial machine learning. It mainly reviews the attack and defense methods in the field of deep learning. And several new research directions, such as generative adversarial networks, are presented. The adversarial attack strategies in complex scenarios, such as reinforcement learning and physical attack, are also briefly recommended.	This paper mainly introduces the problem of image classification in the field of computer vision. There are few researches in the field of text and malware in the article. The research on the nonconvolutional structure is less, either.
Naveed Akhtar et al. 2018 [127]	This paper reviews the adversarial attack work of deep learning in computer vision. It mainly introduces the adversarial attack and defensive measures under deep learning. In addition, the security application literature provides a broader prospect for the research direction of adversarial attack in retrospect.	It mainly introduces the serious threat to deep learning models caused by perturbations to images in the field of computer vision. Examples of creating adversarial samples for text and malware classification are only briefly mentioned.
Shilin Qiu et al. 2019 [128]	This paper summarizes the latest research progress of adversarial attack and defense techniques in deep learning. It mainly reviews the adversarial attack of the target model in the training stage and the test stage and concludes the application of adversarial attack in the four fields of image, text, cyber space security, and physical world as well as the existing defense methods.	The adversarial attack is not analyzed in terms of the result of the attack.
Wei Emma Zhang et al. 2019 [129]	This paper is the first to make a comprehensive summary of the text deep neural network model adversarial attacks. It mainly reviews the adversarial attack and deep learning in the field of natural language processing, briefly introduces the defense methods, and discusses some open issues.	In this paper, the research of generating textual adversarial examples on DNNs in the field of natural language processing are summarized. However, it seldom introduces the architecture of deep neural network.
Wei Jin et al. 2020 [98]	This paper gives a comprehensive introduction to the research of graph neural network adversarial attack algorithm classification and defense strategy classification. The performance of different defense methods under different attacks is also empirically studied, and a repository of representative algorithms is developed.	It summarizes the adversarial attack and defense technology of graphic data but does not introduce the adversarial attack of other types of data.
Wenqi Wang et al. 2020 [130]	This essay comprehensively summarizes the research of textual adversarial examples in different fields. It mainly concludes the attack and defense classification of DNN in the text. Also, it discusses how to build a robust DNN model through testing and validation.	The adversarial attack and defense techniques in the field of images and malicious code are not analyzed.
Han Xu et al. 2020 [131]	This paper reviews the attack and defense methods on DNN models for image, graph, and text data. The algorithms and defense strategies for generating adversarial samples of three types of data are reviewed.	From the prospective of application field, image classification and natural language processing are introduced comprehensively, while malicious code detection is only briefly mentioned.
Jiliang Zhang et al. 2020 [132]	This paper gives a comprehensive summary of the existing adversarial example generation methods. This paper mainly introduces the basic concept of adversarial example, the comparison of different adversarial attack methods and defensive measures.	Like most reviews, this paper introduces and compares several typical attack algorithms such as L-BFGS, FGSM, and C&W. There is no introduction to adversarial attacks in the text and malicious code domains.

(1) This paper follows the route of "Why? → What? → How?" and does not put forward any new concepts, intended to let beginners quickly enter the field of adversarial attack and defense under the guidance of this essay

(2) This paper did not carry out adversarial attack studies from the L-BFGS method, FGSM-based attack, basic and least-likely-class iterative methods, JSMA attack, C&W method, and DeepFool method, as most of the review did. Instead, the research methods of adversarial attack are classified from the fields of image, text, and malicious code to assist researchers to discover the breakthrough point for their research

(3) This paper is a systematic introduction of influential papers published after 2010 to ensure the advanced and comprehensive of the essay. Researchers can quickly find their interest points by browsing this article, improving the efficiency of the study.

8. Conclusions

In the field of AI security, it is a constant battle between attack and defense. To help researchers quickly enter the field of adversarial attack, this review is based on high-quality articles published since 2010. We summarize the typical adversarial attacks in the fields of text, images, and malware to help researchers locate their own research areas. Also, we introduce defense technologies against attacks. Finally, we present some discussions and open issues. Adversarial learning has a long history in the field of security. It is hoped that under the guidance of this paper, new researchers can effectively establish the framework of adversarial attack and defense.

Data Availability

Previously published articles were used to support this study, and these prior studies and datasets are cited at relevant places within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Funding

This work was supported by the National Key Research & Development Program of China (2020YFB1712104), Major Scientific and Technological Innovation Projects of Shandong Province (2020CXGC010116), the National Natural Science Foundation of China (No. 61876019, No. U1936218, No. 62072037), and Zhejiang Lab (No. 2020LE0AB02).

References

[1] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "STRIP: a defence against Trojan attacks on deep neural networks," in *Proceedings of the 35th Annual Computer Security Applications Conference*, pp. 113–125, New York, NY, USA, 2019.

[2] B. Biggio and F. Roli, "Wild patterns: ten years after the rise of adversarial machine learning," *Pattern Recognit.*, vol. 84, pp. 317–331, 2018.

[3] C. Esposito, M. Ficco, and B. B. Gupta, "Blockchain-based authentication and authorization for smart city applications," *Information Processing & Management*, vol. 58, no. 2, p. 102468, 2021.

[4] D. Li, L. Deng, B. B. Gupta, H. Wang, and C. Choi, "A novel CNN based security guaranteed image watermarking generation scenario for smart city applications," *Information Sciences*, vol. 479, pp. 432–447, 2019.

[5] C. Szegedy, W. Zaremba, I. Sutskever et al., "Intriguing properties of neural networks," in *2nd International Conference on Learning Representations, ICLR 2014*, Banff, Canada, 2014https://nyuscholars.nyu.edu/en/publications/intriguing-properties-of-neural-networks.

[6] X. Liu, J. Zhang, Y. Lin, and H. Li, "ATMPA: attacking machine learning-based malware visualization detection methods via adversarial examples," in *Proc. IEEE/ACM Int. Symp. Qual. Service*, Phoenix, AZ, USA, June 2019.

[7] M. Zhou, J. Wu, Y. Liu, S. Liu, and C. Zhu, "DaST: data-free substitute training for adversarial attacks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020.

[8] Z. Guan, Z. Lv, X. Sun et al., "A differentially private big data nonparametric Bayesian clustering algorithm in smart grid," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2631–2641, 2020.

[9] J. Dong, Z. Guan, L. Wu, X. Du, and M. Guizani, "A sentence-level text adversarial attack algorithm against IIoT based smart grid," *Computer Networks*, vol. 190, article 107956, 2021.

[10] J. L. Di Wu, S. K. Das, J. Wu, Y. Ji, and Z. Li, "A novel distributed denial-of-service attack detection scheme for software defined networking environments," in *2018 IEEE international conference on communications (ICC)*, Kansas City, MO, USA, 2018.

[11] B. Zhou, J. Li, J. Wu, S. Guo, Y. Gu, and Z. Li, "Machine-learning-based online distributed denial-of-service attack detection using spark streaming," in *IEEE International Conference on Communications (ICC)*, Kansas City, MO, USA, 2018.

[12] W. Han, J. Xue, Y. Wang, S. Zhu, and Z. Kong, "Review: build a roadmap for stepping into the field of anti-malware research smoothly," *IEEE Access*, vol. 7, pp. 143573–143596, 2019.

[13] N. Carlini, G. Katz, C. Barrett, and D. Dill, "Ground-truth adversarial examples," 2017, https://arxiv.org/abs/1709.10207.

[14] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning: passive and active white-box inference attacks against centralized and federated learning," in *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 739–753, San Francisco, CA, USA, 2019.

[15] W. Xiao, H. Jiang, and S. Xia, "A new black box attack generating adversarial examples based on reinforcement learning," in *2020 Information Communication Technologies Conference (ICTC)*, pp. 141–146, Nanjing, China, 2020.

[16] Z. Zhou, B. Wang, M. Dong, and K. Ota, “Secure and efficient vehicle-to-grid energy trading in cyber physical systems: integration of blockchain and edge computing,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 43–57, 2020.

[17] Z. Katzir and Y. Elovici, “Why blocking targeted adversarial perturbations impairs the ability to learn,” 2019, https://arxiv.org/abs/1907.05718.

[18] A. Wu, Y. Han, Q. Zhang, and X. Kuang, “Untargeted adversarial attack via expanding the semantic gap,” in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 514–519, Shanghai, China, 2019.

[19] H. S. Anderson, A. Kharkar, B. Filar, D. Evans, and P. Roth, “Learning to evade static PE machine learning malware models via reinforcement learning,” 2018, https://arxiv.org/abs/1801.08917.

[20] A. Demontis, M. Melis, M. Pintor et al., “Why do adversarial attacks transfer? Explaining transferability of evasion and poisoning attacks,” in *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pp. 321–338, USA, 2019.

[21] B. Wang, Y. Yao, S. Shan et al., “Neural cleanse: identifying and mitigating backdoor attacks in neural networks,” in *2019 IEEE Symposium on Security and Privacy (SP),*, pp. 707–723, San Francisco, CA, USA, 2019.

[22] T. D. Nguyen, P. Rieger, M. Miettinen, and A.-R. Sadeghi, “Poisoning attacks on federated learning-based iot intrusion detection system,” in *Workshop on Decentralized IoT Systems and Security (DISS) @ NDSS Symposium 2020*, pp. 23–26.02, San Diego, USA, 2020.

[23] J. Monteiro, Z. Akhtar, and T. Falk, “Generalizable adversarial examples detection based on bi-model decision mismatch,” in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 2839–2844, Bari, Italy, 2019.

[24] N. Papernot, P. McDaniel, A. Swami, and R. E. Harang, “Crafting adversarial input sequences for recurrent neural networks,” in *MILCOM 2016-2016 IEEE Military Communications Conference*, pp. 49–54, Baltimore, MD, USA, 2016.

[25] D. Jin and Z. Jin, *Text Fool: Fool your Model with Natural Adversarial Text*, 2019.

[26] J. Ebrahimi, A. Rao, D. Lowd, and D. Dou, “Hot flip: white-box adversarial examples for text classification,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 31–36, Melbourne, Australia, 2018.

[27] M. Alzantot, Y. Sharma, A. Elgohary, B.-J. Ho, M. Srivastava, and K.-W. Chang, “Generating natural language adversarial examples,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, 2018.

[28] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, “Black-box generation of adversarial text sequences to evade deep learning classifiers,” in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 50–56, San Francisco, CA, USA, 2018.

[29] Z. Zhao, D. Dua, and S. Singh, “Generating natural adversarial examples,” 2018, https://arxiv.org/abs/1710.11342.

[30] J. Li, S. Ji, T. Du, B. Li, and T. Wang, “Text bugger: generating adversarial text against real-world applications,” in *Proceedings 2019 Network and Distributed System Security Symposium*, San Diego, CA, USA, 2019.

[31] Y. Gil, Y. Chai, O. Gorodissky, and J. Berant, “White-to-black: efficient distillation of black-box adversarial attacks,” in *Proceedings of the 2019 Conference of the North*, Minneapolis, Minnesota, 2019.

[32] S. Das and P. N. Suganthan, “Differential evolution: a survey of the state-of-the-art,” *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.

[33] D. C. Liu and J. Nocedal, “On the limited memory BFGS method for large scale optimization,” *Mathematical Programming*, vol. 45, no. 1-3, pp. 503–528, 1989.

[34] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” 2014, https://arxiv.org/abs/1412.6572.

[35] N. Papernot, P. Mcdaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, “The limitations of deep learning in adversarial settings,” in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 372–387, Saarbruecken, Germany, 2015.

[36] S. M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, “DeepFool: a simple and accurate method to fool deep neural networks,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2574–2582, Las Vegas, NV, USA, 2016.

[37] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” in *Artificial Intelligence Safety and Security*, pp. 99–112, OpenReview.net, Toulon, France, 2016.

[38] J. Su, D. V. Vargas, and K. Sakurai, “One pixel attack for fooling deep neural networks,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 828–841, 2019.

[39] N. Carlini and D. Wagner, “Towards evaluating the robustness of neural networks,” in *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, San Jose, CA, USA, 2017.

[40] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, “Universal adversarial perturbations,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),*, pp. 86–94, Honolulu, HI, USA, 2017.

[41] S. Sarkar, A. Bansal, U. Mahbub, and R. Chellappa, “UPSET and ANGRI : breaking high performance image classifiers,” 2017, https://arxiv.org/abs/1707.01159.

[42] M. Cisse, Y. Adi, N. Neverova, and J. Keshet, “Houdini: fooling deep structured visual and speech recognition models with adversarial examples,” *Advances in neural information processing systems*, vol. 30, 2017.

[43] S. Baluja and I. Fischer, “Adversarial transformation networks: learning to generate adversarial examples,” 2017, https://arxiv.org/abs/1703.09387.

[44] W. Hu and Y. Tan, “Generating adversarial malware examples for black-box attacks based on GAN,” 2017, https://arxiv.org/abs/1702.05983.

[45] A. Al-Dujaili, A. Huang, E. Hemberg, and U. M. O’Reilly, “Adversarial deep learning for robust detection of binary encoded malware,” in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 76–82, San Francisco, CA, USA, 2018.

[46] B. Kolosnjaji, A. Demontis, B. Biggio et al., “Adversarial malware binaries: evading deep learning for malware detection in executables,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 533–537, Rome, Italy, 2018.

[47] W. Song, X. Li, S. Afroz, D. Garg, D. Kuznetsov, and H. Yin, “Automatic generation of adversarial examples for interpreting malware classifiers,” 2020, https://arxiv.org/abs/2003.03100.

[48] I. Rosenberg, A. Shabtai, Y. Elovici, and L. Rokach, “Query-efficient black-box attack against sequence-based malware

classifiers," in *Annual Computer Security Applications Conference*, pp. 611–626, Austin, TX, USA, 2020.

[49] M. Ebrahimi, N. Zhang, J. Hu, M. T. Raza, and H. Chen, "Binary black-box evasion attacks against deep learning-based static malware detectors with adversarial byte-level language model," 2020, https://arxiv.org/abs/2012.07994.

[50] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: real and stealthy attacks on state-of-the-art face recognition," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1528–1540, Vienna, Austria, 2016.

[51] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519, Abu Dhabi, United Arab Emirates, 2017.

[52] A. Shafahi, W. R. Huang, M. Najibi et al., "Poison frogs! Targeted clean-label poisoning attacks on neural networks," 2018, https://arxiv.org/abs/1804.00792.

[53] Y. Mirsky, T. Mahler, I. Shelef, and Y. Elovici, "CT-GAN: malicious tampering of 3D medical imagery using deep learning," in *28th {USENIX} Security Symposium ({USENIX} Security 19*, pp. 461–478, USA, 2019.

[54] S. T. Chen, C. Cornelius, J. Martin, and D. H. P. Chau, *Shape Shifter: Robust Physical Adversarial Attack on Faster R-CNN Object Detector: Recognizing Outstanding, [Ph.D. thesis]*, Springer, 2019.

[55] Q. Xiao, Y. Chen, C. Shen, Y. Chen, and K. Li, "Seeing is not believing: camouflage attacks on image scaling algorithms," *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pp. 443–460, USENIX Association, Santa Clara, CA, 2019.

[56] Y. Wang, Y. Tan, W. Zhang, Y. Zhao, and X. Kuang, "An adversarial attack on DNN-based black-box object detectors," *Journal of Network and Computer Applications*, vol. 161, p. 102634, 2020.

[57] J. Solano, C. Lopez, E. Rivera, A. Castelblanco, L. Tengana, and M. Ochoa, "SCRAP: synthetically composed replay attacks vs. adversarial machine learning attacks against mouse-based biometric authentication," in *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security*, pp. 37–47, 2020.

[58] M. Cheng, J. Yi, P. Y. Chen, H. Zhang, and C. J. Hsieh, "Seq2-Sick: evaluating the robustness of sequence-to-sequence models with adversarial examples," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 4, pp. 3601–3608, 2020.

[59] H. Zhang, H. Zhou, N. Miao, and L. Li, "Generating fluent adversarial examples for natural languages," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019.

[60] Y. Zang, F. Qi, C. Yang et al., "Word-level textual adversarial attacking as combinatorial optimization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2019.

[61] E. Raff, J. Barker, J. Sylvester, R. Brandon, B. Catanzaro, and C. Nicholas, "Malware detection by eating a whole EXE," 2017, https://arxiv.org/abs/1710.09435.

[62] L. Demetrio, B. Biggio, G. Lagorio, F. Roli, and A. Armando, "Functionality-preserving black-box optimization of adversarial windows malware," 2020, https://www.semanticscholar.org/paper/32ff17fb274e7c455587c30cc092d42ccce53a80.

[63] X. Chen, C. Li, D. Wang et al., "Android HIV: a study of repackaging malware for evading machine-learning detection," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 987–1001, 2020.

[64] D. Hendrycks and K. Gimpel, "Early methods for detecting adversarial images," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, pp. 1–9, Toulon, France, 2017.

[65] J. Zhang, S. Peng, Y. Hu et al., "HRAE: hardware-assisted randomization against adversarial example attacks," in *2020 IEEE 29th Asian Test Symposium (ATS)*, Penang, Malaysia, 2020.

[66] "HUAWEI, 'ai-security-white-paper-cn'," http://huawei.com/-/media/corporate/pdf/cyber-security/ai-security-white-paper-cn.pdf.

[67] J. Wang, T. Zhang, S. Liu et al., *Beyond Adversarial Training: Min-Max Optimization in Adversarial Attack and Defense*, 2019, http://arxiv.org/abs/1906.03563.

[68] J. An and D. Wagner, "E-ABS: extending the analysis-by-synthesis robust classification model to more complex image domains," in *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security*, pp. 25–36, 2020.

[69] Y. Chen, S. Wang, D. She, and S. Jana, "On training robust PDF malware classifiers," in *29th USENIX Security Symposium (USENIX Security 20)*, pp. 2343–2360, 2020, https://www.usenix.org/conference/usenixsecurity20/presentation/chen-yizheng.

[70] Z. Zhang and T. Wu, "Adversarial distillation for ordered top-k attacks," 2019, https://arxiv.org/abs/1905.10695.

[71] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE symposium on security and privacy (SP)*, pp. 582–597, San Jose, CA, USA, 2016.

[72] A. G. Amato, "Adversarial examples detection in features distance spaces: subvolume B," in *European Conference on Computer Vision*, Munich, Germany, 2019.

[73] I. Shumailov, Y. Zhao, R. Mullins, and R. Anderson, "Towards certifiable adversarial sample detection," in *Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security*, pp. 13–24, USA, 2020.

[74] S. Gu and L. Rigazio, "Towards deep neural network architectures robust to adversarial examples," 2015, https://arxiv.org/abs/1412.5068.

[75] Y. G. Qian, X. M. Zhang, B. Wang et al., "Towards robust DNNs: a Taylor expansion-based method for generating powerful adversarial examples," 2020, https://arxiv.org/abs/2001.08389.

[76] Y. Shi and Y. Han, "Schmidt: image augmentation for black-box adversarial attack," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, San Diego, USA, 2018.

[77] R. Laishram and P. V. V. Curie, *A Method for Protecting SVM Classifier from Poisoning Attack*, CoRR, 2016.

[78] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li, "Manipulating machine learning: poisoning attacks and countermeasures for regression learning," in *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 19–35, San Francisco, CA, USA, 2018.

[79] D. Li and Q. Li, "Adversarial deep ensemble: evasion attacks and defenses for malware detection," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3886–3900, 2020.

[80] C. Liao, Y. Cheng, C. Fang, and J. Shi, "Where does the robustness come from?: a study of the transformation-based ensemble defence," in *Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security*, pp. 1–12, USA, 2020.

[81] S. KIM and H. Kim, "Zero-centered fixed-point quantization with iterative retraining for deep convolutional neural network-based object detectors," *IEEE Access*, vol. 9, pp. 20828–20839, 2021.

[82] Y. Liu, X. Yang, and A. Srivastava, "Neural Trojans," in *2017 IEEE 35th International Conference on Computer Design (ICCD)*, Boston, MA, USA, 2017.

[83] L. Kang, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: defending against backdooring attacks on deep neural networks," in *International Symposium on Research in Attacks, Intrusions, and Defenses*, Springer, Cham, 2018.

[84] N. Papernot, M. Abadi, U. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," 2016, https://arxiv.org/abs/1610.05755.

[85] M. Abadi, A. Chu, I. Goodfellow et al., "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016https://dl.acm.org/doi/10.1145/2976749.2978318.

[86] J. Giraldo, A. Cardenas, M. Kantarcioglu, and J. Katz, "Adversarial classification under differential privacy," in *Network and Distributed System Security Symposium*, San Diego, California, 2020.

[87] Y. Adi, C. Baum, M. Cisse, B. Pinkas, and J. Keshet, "Turning your weakness into a strength: watermarking deep neural networks by backdooring," in *27th {USENIX} Security Symposium ({USENIX} Security 18)*, USENIX Association, USA, 2018.

[88] H. Yakura, S. Shinozaki, R. Nishimura, Y. Oyama, and J. Sakuma, "Neural malware analysis with attention mechanism," *Computers & Security*, vol. 87, pp. 101592.1–101592.15, 2019.

[89] G. Machado, R. Goldschmidt, and E. Silva, "MultiMagNet: a non-deterministic approach based on the formation of ensembles for defending against adversarial images," in *21st International Conference on Enterprise Information Systems*, Heraklion, Crete, Greece, 2019.

[90] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu, "Defense against adversarial attacks using high-level representation guided denoiser," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 2018.

[91] Z. Yi, J. Yu, S. Li, Y. Tan, and Q. Wu, "Incremental learning of GAN for detecting multiple adversarial attacks," in *Lecture Notes in Computer Science*, Springer, 2019.

[92] S. Ma, Y. Liu, G. Tao, W.-C. Lee, and X. Zhang, "NIC: detecting adversarial samples with neural network invariant checking," in *Presented at the Network and Distributed System Security Symposium*, San Diego, CA, 2019.

[93] X. Ling, S. Ji, J. Zou et al., "DEEPSEC: a uniform platform for security analysis of deep learning model," in *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 673–690, San Francisco, CA, USA, May 2019.

[94] M. Fang, X. Cao, J. Jia, and N. Gong, "Local model poisoning attacks to byzantine-robust federated learning," *29th {USENIX} Security Symposium ({USENIX} Security 20)*, USENIX Association, 2019.

[95] Z. Yang, J. Zhang, E.-C. Chang, and Z. Liang, "Neural network inversion in adversarial setting via background knowledge alignment," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 225–240, London United Kingdom, 2019.

[96] P. Korshunov and S. Marcel, "Deep Fakes: a new threat to face recognition? Assessment and detection," 2018, https://arxiv.org/abs/1812.08685.

[97] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, and B. Y. Zhao, "Fawkes: protecting privacy against unauthorized deep learning models," *29th {USENIX} Security Symposium ({USENIX} Security 20)*, USENIX Association, 2020.

[98] W. Jin, Y. Li, H. Xu, Y. Wang, and J. Tang, "Adversarial attacks and defenses on graphs: a review and empirical study," 2020, https://arxiv.org/abs/2003.00653.

[99] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deep face: closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708, Columbus, OH, USA, 2014.

[100] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015, https://arxiv.org/abs/1409.1556.

[101] Z. A. Din, H. Venugopalan, and J. Park, "Boxer: preventing fraud by scanning credit cards," in *29th USENIX Security Symposium (USENIX Security 20)*, pp. 1571–1588, USENIX Association, 2020, https://www.usenix.org/conference/usenixsecurity20/presentation/din.

[102] H. Yu, K. Yang, T. Zhang, Y.-Y. Tsai, T.-Y. Ho, and Y. Jin, "Cloud leak: large-scale deep learning models stealing through adversarial examples," in *presented at the Network and Distributed System Security Symposium*, San Diego, CA, 2020.

[103] M. Sato, J. Suzuki, H. Shindo, and Y. Matsumoto, "Interpretable adversarial perturbation in input embedding space for text," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, Stockholm, 2018.

[104] P. Neekhara, S. Hussain, S. Dubnov, and F. Koushanfar, "Adversarial reprogramming of text classification neural networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, 2018https://arxiv.org/abs/1809.01829.

[105] D. Li, D. V. Vargas, and K. Sakurai, "Universal rules for fooling deep neural networks based text classification," in *2019 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2221–2228, Wellington, New Zealand, 2019.

[106] Y. Huang, U. Verma, C. Fralick, G. Infante-Lopez, B. Kumar, and C. Woodward, "Malware evasion attack and defense," in *2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pp. 34–38, Portland, OR, USA, 2019.

[107] O. Suciu, S. E. Coull, and J. Johns, "Exploring adversarial examples in malware detection," in *2019 IEEE Security and Privacy Workshops (SPW)*, pp. 8–14, San Francisco, CA, USA, 2019.

[108] P. Xu, B. Kolosnjaji, C. Eckert, and A. Zarras, "MANIS: evading malware detection system on graph structure," in

Proceedings of the 35th Annual ACM Symposium on Applied Computing, New York, NY, USA, 2020.

[109] D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, and K. Rieck, "Drebin: effective and explainable detection of Android malware in your pocket," in *Presented at the Network and Distributed System Security Symposium*, San Diego, CA, 2014.

[110] S. Zhang, X. Xie, and Y. Xu, "A brute-force black-box method to attack machine learning-based systems in cybersecurity," *IEEE Access*, vol. 8, pp. 128250–128263, 2020.

[111] M. Alaeiyan and S. Parsa, "Detection of algorithmically-generated domains: an adversarial machine learning approach," *Computer Communications*, vol. 160, pp. 661–673, 2020.

[112] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, "ML-Leaks: model and data independent membership inference attacks and defenses on machine learning models," in *presented at the Network and Distributed System Security Symposium*, San Diego, CA, 2019.

[113] J. Yan, Y. Qi, and Q. Rao, "Detecting malware with an ensemble method based on deep neural network," *Security and Communication Networks*, vol. 2018, Article ID 7247095, 16 pages, 2018.

[114] J. W. Stokes, D. Wang, M. Marinescu, M. Marino, and B. Bussone, "Attack and defense of dynamic analysis-based, adversarial neural malware detection models," in *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*, pp. 1–8, Los Angeles, CA, USA, 2018.

[115] "scikit-learn," https://scikit-learn.org.cn/.

[116] Y. A. N. Jia, Y. A. N. Jia, N. I. E. Chujiang, and S. U. Purui, "Method for generating malicious code adversarial samples based on genetic algorithm," *Journal of Electronics and Information Technology*, vol. 42, no. 9, pp. 2126–2133, 2020.

[117] H. Krawczyk, "HMQV: a high-performance secure Diffie-Hellman protocol," in *Advances in Cryptology-CRYPTO 2005: 25th Annual International Cryptology Conference*, Santa Barbara, California, USA, 2005.

[118] D. Wang, W. Li, and P. Wang, "Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4081–4092, 2018.

[119] S. Qiu, D. Wang, G. Xu, and S. Kumari, "Practical and provably secure three-factor authentication protocol based on extended chaotic-maps for mobile lightweight devices," *IEEE Transactions on Dependable and Secure Computing*, p. 1, 2020.

[120] D. Wang and P. Wang, "Two birds with one stone: two-factor authentication with security beyond conventional bound," *IEEE Transactions on Dependable & Secure Computing*, vol. 15, no. 4, pp. 708–722, 2018.

[121] S. Qiu, G. Xu, H. Ahmad, and L. Wang, "A robust mutual authentication scheme based on elliptic curve cryptography for Telecare medical information systems," *IEEE Access*, vol. 6, pp. 7452–7463, 2018.

[122] W. Zheng, R. A. Popa, J. E. Gonzalez, and I. Stoica, "Helen: maliciously secure coopetitive learning for linear models," in *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 724–738, San Francisco, CA, USA, May 2019.

[123] M. Pawlicki, M. Chora, and R. Kozik, "Defending network intrusion detection systems against adversarial evasion attacks," *Future Generation Computer Systems*, vol. 110, no. 11, pp. 148–154, 2020.

[124] W. Li, Y. Wang, J. Li, and M. H. Au, "Toward a blockchain-based framework for challenge-based collaborative intrusion detection," *International Journal of Information Security*, vol. 20, no. 2, pp. 127–139, 2021.

[125] W. Meng, E. W. Tischhauser, Q. Wang, Y. Wang, and J. Han, "When intrusion detection meets blockchain technology: a review," *IEEE Access*, vol. 6, pp. 10179–10188, 2018.

[126] G. Li, P. Zhu, J. Li, Z. Yang, N. Cao, and Z. Chen, "Security matters: a survey on adversarial machine learning," 2018, https://www.semanticscholar.org/paper/6ede8b02b817a1354b8bde1ab7af07b0ddb02acf.

[127] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: a survey," *IEEE Access*, vol. 6, pp. 14410–14430, 2018.

[128] S. Qiu, Q. Liu, S. Zhou, and C. Wu, "Review of artificial intelligence adversarial attack and defense technologies," *Applied Sciences*, vol. 9, no. 5, p. 909, 2019.

[129] W. E. Zhang, Q. Z. Sheng, A. Alhazmi, and C. Li, "Adversarial attacks on deep learning models in natural language processing: a survey," 2019, https://arxiv.org/abs/1901.06796.

[130] W. Wang, L. Wang, R. Wang, Z. Wang, and A. Ye, "Towards a robust deep neural network in texts: a survey," 2020, https://arxiv.org/abs/1902.07285.

[131] H. Xu, Y. Ma, H. C. Liu et al., "Adversarial attacks and defenses in images, graphs and text: a review," *International Journal of Automation and Computing*, vol. 17, no. 2, pp. 151–178, 2020.

[132] J. Zhang and C. Li, "Adversarial examples: opportunities and challenges," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 7, pp. 2578–2593, 2020.