Technium Vol. 12, pp.65-83 (2023)
ISSN: 2668-778X
www.techniumscience.com

A review of Deep Learning Privacy, Security and Defenses

Afrah Salman Dawood [1], Noor Kadhim Hadi [1]
[1] University of Technology- Iraq

Abstract: Deep learning (DL) can be considered as a powerful tool in different fields and for different applications but its importance raised the concern about privacy, security, and defense issues. This research presents an important overview about different aspects and state-of-the-art techniques in DL privacy, security, and defense. Wide range of topics was covered including private data frameworks, different types of threats and attacks, and the most important defense techniques. We have also discussed the challenges and limitations of each approach besides to possible future research directions. This survey can be considered as a comprehensive guide for other researchers and policymakers who are interested in understanding these important topics associated with DL.

Keywords: Deep Learning, Adversarial Attack, Private Framework, Classification Attacks, DeepSecure, adversarial defenses.

I. Introduction

DL, also referred to as hierarchical learning and deep-structured learning, encompasses both supervised and unsupervised machine learning techniques. It was inspired by the design and operation of the human brain, as well as the way neurons interpret messages. Like Artificial Neural Networks (ANNs), DL also includes input, output, and multiple hidden layers. The response of each DL layer is dependent on the nonlinear response generated by the input layer using the data it receives. DL has been extensively used in various domains such as speech recognition, image recognition, object detection, drug discovery for illnesses and genomes, etc., over the last few years. By utilizing backpropagation, DL models adjust the parameters of their layers iteratively, optimizing them to minimize the difference between predicted and actual outcomes. This training process enables DL models to capture high-level abstractions and extract valuable insights from raw data, making them highly effective in tasks such as image and speech recognition, natural language processing, and more [1] [2], [3] [4].

The DL's encrypted data, which comes from training and interface modules, is its main issue. Due to the widespread adoption of DL models in various applications, as previously indicated, security and privacy concerns are of utmost importance. Furthermore, the training portion of all models using DL really depends on a sizable amount of large data, sensitive data, and personal user data, particularly training data. In light of this, DL models must not provide sensitive and private information [5], [6].

Cybersecurity refers to a collection of laws, techniques, tools, and processes that work together to protect computer systems, networks, software, and data from unauthorized access while maintaining their availability, confidentiality, and integrity. To prevent cyber-attacks, security measures are implemented at various levels, including applications, networks, hosts, and data. Different security solutions such as firewalls, antivirus software, intrusion detection systems (IDSs), and intrusion protection systems (IPSs) are used to detect and prevent security breaches. However, cyber attackers often only need to exploit one vulnerability to gain access to a system. As the number of systems connected to the internet increases, the attack surface also grows, making it easier for attackers to launch assaults. Attackers are becoming more sophisticated, using zero-day exploits and malware that can bypass security mechanisms, making it harder

for defenders to detect and prevent attacks. Defenders must also be aware of insider threats from individuals or entities within an organization who misuse their authorized access, in addition to external attacks.[3], [7].

Servers, smart devices, applications and other cyber-enabled resources create enormous amounts of data through interactions between machines and between people and machines [8]–[10]. The Security Information Event Management (SIEM) system, which frequently overwhelms the security analyst with event notifications, is one example of a cyber defense system that generates a lot of data. Data science may be used in cyber security to better any defense program's security posture by correlating events, identifying trends, and spotting aberrant activity. Cyber protection solutions that use data analytics are beginning to appear. Network intrusion detection systems (NIDSs), for example, are changing from signature-based systems that only identify well-known assaults to anomaly-based systems that only detect departures from "typical" behavior profiles. [11]–[13].

II. Deep Learning Private Data Frameworks

This section provides an overview of the top private data security frameworks for DL. These frameworks are highly secure under the Honest-but-Curious (HbC) adversary paradigm. The Honest-but-Curious (HbC) adversary paradigm is a security model used in cryptography and secure computation. It describes a scenario in which an entity or participant involved in a secure protocol is assumed to follow the prescribed protocol honestly, faithfully executing the required steps, but also exhibits curiosity by trying to learn additional information beyond what is allowed by the protocol. In the HbC paradigm, the adversary is not malicious or actively trying to disrupt the protocol. Instead, they attempt to gain additional knowledge or extract sensitive information by observing the protocol's execution or by analyzing the outputs received from other participants. The HbC paradigm helps in understanding and designing secure protocols that can withstand adversaries who are honest in protocol execution but curious about obtaining unauthorized information. This protocol is especially secure because it prevents malicious attacks and prohibits participants from violating protocol regulations. Preserving accuracy in machine learning refers to maintaining the performance and predictive power of a model while applying techniques such as data encryption, privacy-preserving algorithms, or secure computation to protect sensitive data during training or inference. Cryptographic protocols are mathematical techniques used to secure communication and computation in scenarios involving sensitive information, ensuring confidentiality, integrity, and authentication while preserving privacy. Constant number of interactions refers to protocols or algorithms that require a fixed and limited number of communication rounds or interactions to complete a computation or secure task, providing efficiency and reducing latency. Nonpolynomial activation functions, such as the rectified linear unit (ReLU), are commonly used in DL to introduce non-linearity and enable the learning of complex patterns, enhancing the model's representational capacity. Max-pooling is a down-sampling technique in DL that reduces the dimensionality of feature maps by selecting the maximum value within each pooling region. It helps extract dominant features while providing translation invariance. These concepts are compared to understand their effectiveness, efficiency, and trade-offs in preserving accuracy, ensuring secure computations, minimizing communication overhead, enhancing model expressiveness, and maintaining robustness against attacks in scenarios where privacy and security are crucial considerations. Table 1 presents a high-level comparison of the frameworks and the resources that support each framework [14], [15] [16].

Technium Vol. 12, pp.65-83 (2023)
ISSN: 2668-778X
www.techniumscience.com

Table 1: DL private data frameworks properties with related cryptographic protocols and related researches.

Framework	Preserving Accuracy	Constant number of interactions	Constant preprocessing	DL model privacy	Scalability for large DL models	Nonpolynomial and max-pooling activation	Cryptographic protocol(s)	Related researches
CryptoNets	×	✓	×	×	×		Leveled-HE	[17]–[19]
SecureML	×	×	×	✓	×		Linearly-HE, GC, SS, MPC	[20]–[23]
MiniONN	×	×	×	✓	×		Additively HE, GC, SS	[24]–[26]
DeepSecure	✓	✓	×	✓	✓		GC	[27]–[29]
Chameleon	✓	×	×	✓	✓		GMW, GC, SS	[26], [30], [31]
Gazelle	✓	×	×	✓	✓		Additively HE, GC, SS	[32]–[35]
XONN	✓	✓	✓	✓	✓		HE, DP, SA, MPC	[36], [37]
Shokri and Shmatikov	×	×	×	×	×		HE, SA, MPC, DP	[38]–[40]
Google	×	×	×	×	×		HE, MPC, DP, SE, SA, FL	[41]–[44]

1. CryptoNets

Cryptonets is a type of DL architecture that allows for the training of neural networks on encrypted data. It's designed to maintain the privacy of the data while still allowing for the model to be trained effectively. The Microsoft Research team primarily created CryptoNets by adding Leveled-HE protocol which enables performing computations on encrypted data with varying levels of complexity, providing a scalable and efficient solution for privacy-preserving computations without requiring decryption, while introducing computational overhead compared to non-HE schemes. The authors suggested employing polynomials of several degrees to approach the activation functions because nonlinear activation functions cannot be accomplished using LHE. In order to maintain high prediction accuracy, the neural network needs to be retrained in plain text with the same activation function. However, this strategy has a downside in that it imposes a limitation on the total number of serial multipliers, making the solution unaffordable. Additionally, to achieve a higher level of anonymity using CryptoNets, accuracy must be sacrificed while maintaining the same computing power [14], [45].

2. SecureML

It is a way to teach neural networks in particular how to ensure privacy in general. The HE, GC, and SS (Secret Sharing) protocols form the foundation of the system. It is worth mentioning that SS is a

cryptographic technique where a secret is divided into shares among multiple parties. The secret can only be reconstructed by combining a sufficient number of shares, providing security even if some parties are compromised. Owners of data covertly transfer their information to servers that train a certain neural network and break the law. SecureML trains a neural network with secure account protocols utilizing a more effective custom activation function [46], [47]. The managed model is afterwards secretly distributed across the servers at the conclusion of the account. SecureML offers training as well as a solution to protect privacy. Authors in [48] presented SIMC (Secure Inference with Multi-Client Collaboration) 2.0, which complies with SecureML's SIMC's fundamental structure while also vastly improving the model's linear and non-linear layers. SecureML's SIMC's fundamental structure is a framework for privacy-preserving machine learning. Its fundamental structure involves multiple clients collaborating to perform secure model inference without disclosing their individual data. The fundamental structure of SIMC in SecureML emphasizes privacy preservation during model inference by leveraging secure Multi-Party Computation (MPC) techniques. It enables multiple clients to collaboratively compute model outputs while keeping their individual data confidential.

3. Miniature Oblivious Neural Network (MiniONN)

The MiniONN is a private data framework for DL that uses ONN to enable secure training and prediction. The purpose of Oblivious Neural Networks is to ensure that the training process does not disclose any sensitive information about the data being used. It provides a technique for converting an existing Deep Neural Network (DNN) into a newly created Oblivious Neural Network, which addresses privacy concerns. This approach claims that neither the client nor the server is aware of the model's input from the client-side. MiniONN is more efficient than other techniques such as CryptoNets and SecureML. It leverages secret sharing, garbled circuits, additive HE, and activation functions like pooling for Convolutional Neural Networks (CNNs) [49], [50]. It consists of two primary stages:

- An offline phase that is non-input dependent and enables additive HE.
- Online phases that use Garbled Circuits (GC) and Secret Sharing (SS) to process data through nonlinear layers.

4. DeepSecure

It is based on the Garbled Circuit protocol and is one of the contemporary frameworks. The framework supports any nonlinear activation functions as garbled circuit is a general function evaluation protocol. In order to compress the account and link up to two items in size, DeepSecure proposes reducing the size of the data and the network before the installation of the Garbled Circuits. The preprocessing stage may be used by any other backend engine for its inference because it is independent of the fundamental encryption protocol. Since the client has limited resources, DeepSecure additionally offers safe outsourcing of the account to a backup server [51], [52].

5. Chameleon

This protocol uses a variety of frameworks to protect privacy. The current work of the GMW protocol for detailed study of the activation function and other Garbled Circuits for complex activation functions and pooling layers are both utilized in this framework. Chameleon does addition and subtraction through secret sharing. Like MiniONN, it features offline and online stages. As opposed to the online phase, the offline computing offered more rapid calculation for predictions. Comparing the Chameleon to the other methods mentioned, it is more effective [53].

6. Gazelle

Another mixed-protocol method is Gazelle15, which recommends utilizing Homomorphic Encryption (HE) protocol for linear operations and Garbled Circuit (GC) protocol for nonlinear activation functions. GC is a cryptographic technique used for secure MPC. It allows multiple parties to collaboratively compute a function on their private inputs while preserving the privacy of those inputs. The protocol involves the creation of garbled circuits, which hide the underlying computation, and the parties can

securely evaluate the circuit to obtain the desired output without revealing their private data. The authors provide an effective HE-based convolutional layer implementation technique. As a result, Gazelle speeds up the execution of private inference and minimizes client-server interactions. Mixed-protocol approaches to private inference runtime reduction are intriguing, but they call for at least one cycle of client-server communication for each layer of the neural network. Due to the large connection delay in Internet settings, this feature might dramatically degrade performance and lengthen execution time [54], [55].

7. XONN

In order to make the neural network's underlying operations more compatible with secure computing protocols, XONN proposes an alternative strategy. To prevent any multiplication during the private inference, the authors suggest binarizing the neural network. In other words, all NN parameters and weights can only take a binary value of either 0 or 1. As a result, the multiplication is now an amalgamation of XOR and bit-count operations. The GC protocol may be used as the back-end cryptographic engine to conduct the XOR operation with little to no computation and no communication. As a result, XONN performs at the cutting edge of private inference. Also, regardless of the quantity of NN layers, XONN only needs a fixed number of communication rounds. Another benefit of relying on standalone GC is that it may be modified to be safe against active attackers who may break the protocol at any point using standard protocols. For secure and privacy-preserving medical diagnosis of breast cancer, diabetes, liver illness, and malaria infection with inference times in the range of tens to several hundred milliseconds, the authors provide experimental results on four medical data sets. We go into further detail about the XONN transformation process in the next section [37], [56]–[58].

8. Google Private Data Framework (GPDF)

GPDF also leverages cryptographic protocols for DL applications. One of the key protocols used in this context is Secure Aggregation (SA), which enables multiple parties to aggregate their models trained on their respective private datasets without compromising the privacy of the data. This protocol uses techniques such as HE and secret sharing to enable SA.

Another protocol used in GPDF for DL is Federated Learning (FL), which enables multiple parties to collaboratively train a DL model on their respective private datasets without sharing the raw data with each other. SE is a protocol with techniques like Differential Privacy (DP) and hardware-based Secure Enclaves (SE) in order to enhance data security and privacy. Some modern processors provide SE feature to allow for the execution of data code and storage in the protected area of the processor that is isolated from the rest of the system. The purpose of this technique is to provide safe space for important and sensitive operations such as encryption and decryption [59], [60]. Another important technique used by GPDF is the cryptography that enables secure inference on encrypted data like HE that can be used for performing encrypted inference on encrypted data without data encryption [31].

Overall, using cryptographic protocols in GPDF for DL applications ensures secure use of private data for training while preserving data privacy. As the need for secure and private DL increases, the use of such protocols will become increasingly important to ensure the protection of sensitive data [3], [61], [62].

9. Shokri and Shmatikov Private Data Framework (SSPDF)

SSPDF can be considered a system that is designed for private DL by allowing multiple parties to train a model on their private datasets without sharing their data. Like GPDF, SSPDF also employs cryptographic protocols to secure the privacy of the data. Here are some of the cryptographic protocols used in SSPDF for DL applications:

• HE: used in SSPDF to enable encrypted computation on the data where parties can perform computations on their encrypted data without the need for decryption.

• DP: used to add random noise to the data to prevent individual data points from being identified. This technique protects sensitive information while allows meaningful analysis.

• Secure MPC: enables multiple parties to jointly compute a function on their private data without revealing the data to each other. It is a useful protocol when multiple parties need to collaborate on data analysis while preserving privacy.
• SA: enables model updates aggregation from multiple parties without revealing individual updates. It uses techniques like HE and secret sharing to enable SA.
A comparison among these types can be illustrated as shown in Table 2. Generally, using cryptographic protocols in SSPDF for DL applications enables the use of secure private data for training while maintaining data privacy. As the need for secure and private DL increases, the use of such protocols will become increasingly important in ensuring that sensitive data remains protected [63], [64]. Each of these frameworks has its own strengths and weaknesses, and the best choice depends on the specific requirements of the use case, the type of data, the level of security required, and the computational resources available.

Table 2: DL frameworks comparison

DL Framework	Key feature
CryptoNets	Operate on encrypted data
SecureML	Focuses on secure MPC
MiniONN	Uses HE
DeepSecure	Uses secure MPC
Chameleon	Uses cryptographic techniques like HE
Gazelle	Combines HE and two-party computation
XONN	Uses HE and other cryptographic techniques
Shokri and Shmatikov	Uses distributed, privacy-preserving approach
Google	Uses techniques like federated learning and DP

III. Deep Learning Threats and Attacks

In the world of computer security, the word "adversary" is used to refer to individuals or devices that could try to get into or corrupt a computer network or software. A machine learning model can be disrupted by adversaries using a number of attack techniques, either during training (referred to as a "poisoning" assault) or after the classifier has already been learned (referred to as a "evasion" attack). Adversarial attacks outside of research labs have been infrequent thus far. However, cybersecurity researchers are concerned that adversarial attacks could become a significant issue in the future as machine learning is integrated into a wider range of systems, such as self-driving cars and other technologies where human lives could be at risk.

A) Security threats [65]–[68]

1. DL threat type-I:

The most common vulnerabilities in DL frameworks are programming errors that can lead to software failures, infinite loops, or memory exhaustion resulting in the system becoming unresponsive. Denial-of-service (DoS) attacks targeting programs running on top of the window pose the most immediate threat from these issues. While DL models themselves may not be directly harmed by DoS attacks, their performance and availability can be significantly impacted by resource exhaustion (i.e., hardware

components) for example by repeatedly request predictions from a DL model hosted on a server, data poisoning (like flooding the DL model with malicious or incorrect data). There is also the model inference DoS by sending large number of complex input samples to take a long processing time). Another impact comes from disruption of training by interrupting the training process.

2. DL threat type-II:

During the testing phase, deep neural networks are susceptible to attacks. For example, in image recognition, an attacker may test a sample by adding a small amount of noise so that the error is mislabeled by the DNN. This is known as an adversarial example or evasion attack, which can limit the effectiveness of critical security and safety applications such as autonomous cars. Adversarial training is less effective compared to attacks that cannot be seen during training.

3. DL threat type-III:

Software bugs in computers running DL programs on their operating systems can be hijacked due to remote exploitation and application vulnerabilities. This often occurs when the system is connected to a cloud-based system, and the DL applications are also running on that cloud-based system. The network is used to deliver all input to the DL system.

B) Deep learning adversarial attack types

Despite the success of DL in gaining the attention of the industry, its security and privacy challenges have not received the attention they deserve. This discussion focuses on the attack surface of machine learning and the weaknesses in the implementation of DL. In Fig. 1, an adversarial attack on a DL system is illustrated, which is proposed to detect and classify Alzheimer's brain disease into categories such as Non-demented, Mild-demented, Very Mild-demented, and Moderate-demented.

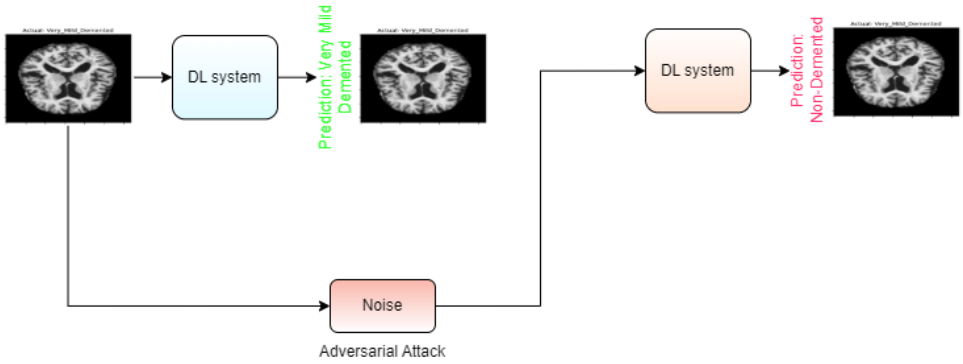

Fig.1: Adversarial attack example

1. Attacks for classification

- Box-constrained LBFGS

L-BFGS [69] stands for Limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm. It's an optimization algorithm in the family of quasi-Newton methods that approximates the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm using a limited amount of computer memory. It's a popular algorithm for parameter estimation in machine learning. L-BFGS is widely used in the machine learning community. It's a good choice for training models where the objective function is differentiable. The box-constrained L-BFGS model was first discovered by Szegedy et al. to solve a box-constrained optimization problem [70]. It was found that the perturbations produced by L-BFGS and applied to the clean pictures may trick the neural network, but the visuals looked real to the human visual system [71].

- Fast gradient sign method (FGSM)

The FGSM (Fast Gradient Sign Method) was introduced by Kurakin et al. [71] to enhance the effectiveness of adversarial training. This method is capable of generating adversarial examples that are not targeted towards any specific output and can be further developed into an iterative approach for

targeted and untargeted attacks. They improved the FGSM method by using the "one-step target class" to generate adversarial examples from original images. FGSM has been widely used to generate adversarial examples for attacking CNN-based or DL image classifiers. It is reported that the top-1 error rate on the adversarial examples generated by FGSM is around 63-69% for ImageNet.

- Basic and Least-Likely-class iterative methods

The one-step approach modifies images by making a single significant movement in the direction that amplifies the loss of the classifier, using a method known as one-step gradient ascent. The goal of this approach is to take several small steps iteratively and adjust the direction after each step.

Other types of adversarial attacks include: One pixel attack, Carlini and wagner attacks (C&W), Deepfool, Upset and angri, Houdini, Adversarial transformation networks (ATNs), Miscellaneous attacks, Targeted universal adversarial attacks [72] [73].

2. Attacks beyond classification/ recognition

In this part, a summary of the research that focuses on developing techniques for attacking deep NNs in applications beyond image classification will be discussed.

- Attacks on autoencoders and generative models

Tabacof et al. [74] published a research on adversarial attacks on autoencoders for image data and a technique for modifying input images to create adversarial examples that deceive the autoencoder into reconstructing a completely different image was provided. Their approach aims to manipulate the internal representation of the neural network, making the adversarial image representation similar to that of the target image. However, in comparison to conventional classifier networks, autoencoders are found to be relatively more robust against adversarial attacks [75].

- Attack on recurrent neural networks

Papernot et al. [76] adversarial input sequences for recurrent neural networks were successfully produced (RNNs). RNNs are DL models that are particularly well suited for learning input-output mappings. Papernot et al. [76] shown that it is possible to trick RNNs using the same techniques used to generate adversarial instances for feed-forward neural networks (such as FGSM) [77]–[79].

- Attacks on deep reinforcement learning

There has been much research on adversarial assaults on traditional DL systems and algorithms, and several different responses have been suggested. Unfortunately, little research has been done on the likelihood and viability of such assaults against Deep Reinforcement Learning (DRL). Designing efficient adversarial assaults is a crucial precondition for creating reliable DRL algorithms, as DRL has demonstrated remarkable success in a variety of challenging tasks [80]. Authors in [81] created two unique adversarial attack approaches to covertly and effectively assault the DRL agents. These two methods give an enemy the ability to introduce hostile samples at the least number of crucial times while still doing the most amount of harm to the agent. The first tactic is a critical point assault, in which the adversary creates a model to forecast future environmental conditions and agent behavior, evaluates the potential harm from several attack strategies, and then chooses the most effective one. The antagonist automatically picks out the crucial moments to attack the agent in an episode using a model that is independent of any particular domain.

- Attacks on semantic segmentation and object detection

Semantic image segmentation and object detection are two of the most popular challenges in the field of Computer Vision. Authors in [82] goal to offer strong defenses against hostile attack on the semantic segmentation paradigm. To do this, they suggest DAPAS, a denoise autoencoder that successfully eliminates adversarial perturbation and prevents an adversarial assault in semantic segmentation. In order for the recovered picture to provide the desired semantic segmentation result, it is crucial to restore the original image at the pixel level since semantic segmentation includes the categorization of pixels. They employ random noise with a specific distribution. They employ the bimodal, uniform, and gaussian

distributions respectively. The adversarial assault would slightly alter the input X pixel value, and the random noise may include several attack strategies. Fig. 2 below shows how CNN model can obtain accurate semantic segmentation results for the original image while adding adversarial noise seriously affects the segmentation effect [83].

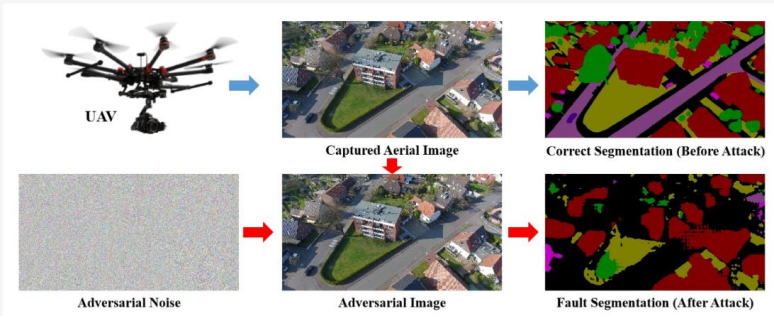

Fig. 2: Semantic segmentation and object detection attack

- Attacks on face attributes

The vulnerability of face recognition to hostile face pictures is well recognized. Previous works create antagonistic pictures by arbitrarily altering a single property without understanding the inherent characteristics of the images. To this end, authors in [84] SAAStarGAN is a novel Semantic Adversarial Attack that tampers with the key face characteristics for each picture. By using the cosine similarity or probability score, they can anticipate the characteristics that matter the most. In order to determine a class probability score for each attribute, the probability score technique relies on training a Face Verification model for an attribute prediction job. The adversarial transferability will be improved along with the ease and effectiveness with which adversarial face pictures may be created thanks to the prediction process. Fig. 3 demonstrates how eyeglasses with adversarial perturbations can trick a facial recognition system into identifying the faces in the second row as those in the first row [85].

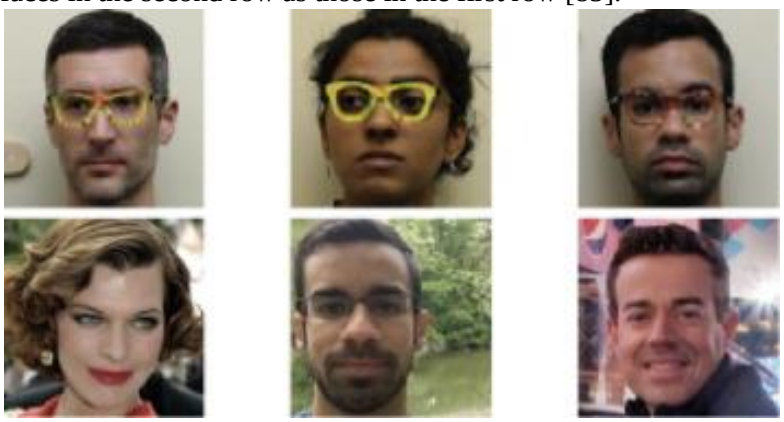

Fig.3: attacks on face attributes.

IV. Deep Learning Defense Techniques

Adversarial training is an effective defense mechanism in certain scenarios that involves adding adversarial examples to the training data of a supervised model to help it recognize them better. This approach aims to reduce the risk posed by adversarial examples by training on both clean and adversarial data. To train a model against adversarial attacks, various techniques are employed, including threat modeling, attack simulation, attack impact evaluation, and countermeasure design. Noise detection is also used for evasion-based attacks. Information laundering is another technique that involves altering the information received by adversaries in model stealing attacks. Finally, to protect algorithms, robust methods are employed to train models.

In general, the representative adversarial defenses are classified into the following:

1. Adversarial training: Intuitive protection against adversarial samples, adversarial training aims to strengthen the resilience of a neural network by training it with adversarial samples. Technically, it is a min–max game where the inner maximization goal is to discover the most effective antagonistic samples, which is achieved by a well-designed adversarial assault, such as FGSM [86] and PGD [87]. The outer minimization is the standard training procedure to minimize the loss. The resulting network is supposed to be resistant against the adversarial attack used for the adversarial sample generation in the training stage.
2. Denoising: For reducing adversarial perturbations/effects, denoising is a relatively simple technique. Two design approaches for such a defense are suggested by prior works: input denoising and feature denoising. The first method seeks to partially or totally eliminate the adversarial perturbations from the inputs, and the second way attempts to ameliorate the impact of adversarial perturbations on high-level features learnt by DNNs [85].
3. Semidefinite programming-based certificated defense [85]: Raghunathan and Kolter [88] First, suggest a verifiable protection strategy for two-layer networks against hostile cases. The authors construct a semidefinite relaxation to upper-bound the adversarial loss and include the relaxation into the training loss as a regularizer. Raghunathan *et al*. [89], further offer a novel semidefinite relaxation for certifying arbitrary ReLU networks. The recently suggested relaxation is more stringent than the prior one and can result in substantial robustness guarantees on three separate networks.
4. Weight-sparse DNNs: Guo *et al*. [90] are the first to demonstrate the intrinsic relationship between weight sparsity and network robustness against FGSM-generated and DeepFool-generated adversarial samples. For linear models Ref. [90] indicates that optimization over adversarial samples might give birth to a sparse solution of the network weights. For nonlinear neural networks, it applies the robustness guarantees from Refs. [91], [92] and indicates that the network Lipchitz constant is prone to be lower when the weight matrices are sparser.
5. KNN-based defenses: The fundamental challenge stems in the fact that identifying an optimum assault on KNN is intractable for ordinary datasets. Authors in Ref. [93] proposed a gradient-based attack on kNN and KNN-based defenses, inspired by the previous work by Sitawarin & Wagner [94]. They demonstrate that our attack outperforms their strategy on all of the models they examined with only a modest increase in the calculation time. With less than 1% of KNN's operating time, the attack also outperforms the most recent attack [95].
6. Bayesian model-based defenses: Authors in paper [96] answers the issue of whether or not model knowledge can help a defender make the right choices when an attacker breaches control system. Using models of the system's stochastic dynamics, the vulnerability that will be exploited, and the attacker's intended outcome, the model-based defense scheme taken into consideration in this study, known as the Bayesian defense mechanism, selects reasonable responses through observation of the system's behavior. On the other hand, logical attackers use deceptive tactics to trick the defense into choosing the wrong course of action. Their dynamic decision-making is described in the cited text as a stochastic signaling game. It is shown that the belief of the true scenario has a limit in a stochastic sense at an equilibrium based on martingale analysis. This fact implies that there are only two possible cases: the defender asymptotically detects the attack with a firm belief, or the attacker takes actions such that the system's behavior becomes nominal after a finite time step. Consequently, if different scenarios result in different stochastic behaviors, the Bayesian defense mechanism guarantees the system to be secure in an asymptotic manner provided that effective countermeasures are implemented. An analysis of a defensive deception using asymmetric recognition of vulnerabilities used by the attacker is done as an application of the discovery. It is proven that the attacker possible ceases the attack even if the defense is ignorant of the exploited vulnerabilities as long as the defender's unawareness is masked by the defensive deceit.

7. Consistency-based defenses: Authors in Ref. [97] They are the first to demonstrate that even context consistency tests may be vulnerable to carefully constructed adversarial scenarios. To develop instances that circumvent these protections, they specifically suggest an adaptive framework, namely Adversarial assaults against object Detection that bypass Context consistency checks (ADC). Ref. [98] proposes PercepGuard as a means of detecting misclassification attacks on perception modules regardless of the attack methodology employed. PercepGuard leverages the spatiotemporal features of a detected object through its tracks to verify that the track and class predictions align. Additionally, to enhance the adversarial resilience of the system against adaptive attacks, context data like ego-vehicle velocity is utilized for verifying contextual consistency, thereby making the attacks more challenging to execute.
8. Randomization: In Ref. [99], the authors classified cyberspace into three domains: physical, network, and digital domains. They designed two agents, one representing the attacker and the other representing the defender, to select actions in the multiple domain cyberspace. The defender's objective is to maximize their reward using reinforcement learning. They proposed a game model based on reward randomization reinforcement learning to improve the defender's defense capabilities. To optimize the defender's strategy and improve the success rate of the defense, the reward is assigned randomly based on a linear distribution when the defender uses multiple domain defense.

V. Discussion

In this study, nine types of frameworks for private data DL processing including CryptoNets, SecureML, MiniONN, etc. have been critically evaluated. The findings suggests that while these frameworks offer promising solutions for preserving DL privacy, they also have significant threats and attacks like DoS, data poisoning and other types explained throughout different sections of the research. Currently, defense techniques like adversarial training, denoising, etc. offer certain levels of protection. However, they also exhibit limitations and can be susceptible to sophisticated attacks. In future, research in DL privacy, security, and defenses should concentrate on the development of more resilient defense techniques and the exploration of innovative strategies for preserving privacy. Specifically, there is a pressing need for more research on protecting DL models from advanced threats such as model inference attacks and disruption of training, with a focus on enhancing the robustness of defenses like adversarial training and randomization defenses. Looking forward, we believe that the next steps in DL privacy, security, and defenses research should focus on developing more robust defense techniques and exploring new approaches to preserving privacy. In particular, more research is needed on how to protect DL models from advanced threats like model inference attacks and disruption of training. Furthermore, as DL models become increasingly complex and are used in more sensitive applications, it will be crucial to develop new frameworks and techniques that can ensure their privacy and security without compromising their performance.

VI. Conclusion

In conclusion, this survey paper provides a comprehensive overview of the current state of DL privacy, security, and defense. The success of DL has led to its widespread adoption in various applications, but it has also raised significant concerns about the privacy and security of individuals and organizations. This work shows that there was a significant research effort in studying these issues like adversarial attacks, DL privacy-preserving, and secure MPC. However, there are many other challenges and limitations to be addressed, and future research directions are needed to overcome them.

References

[1] A. S. Dawood, "Machine Learning and Artificial Neural Network for Data Mining Classification and Prediction of Brain Diseases," *International Journal of Reasoning-based Intelligent Systems*, vol. 1, no. 1, p. 1, 2023, doi: 10.1504/IJRIS.2023.10052940.

[2] B. Shickel, P. Tighe, … A. B.-I. journal of, and undefined 2017, "Deep EHR: a survey of recent advances in deep learning techniques for electronic health record (EHR) analysis," *ieeexplore.ieee.org*, Accessed: Mar. 15, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8086133/

[3] M. I. Tariq *et al.*, "A Review of Deep Learning Security and Privacy Defensive Techniques," *Mobile Information Systems*, vol. 2020, 2020, doi: 10.1155/2020/6535834.

[4] J Schmidhuber, "Deep learning in neural networks: An overview," *Elsevier*, 2015, Accessed: Mar. 15, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608014002135

[5] M. S. Riazi, B. Darvish Rouani, and F. Koushanfar, "Deep Learning on Private Data," *IEEE Secur Priv*, vol. 17, no. 6, pp. 54–63, Nov. 2019, doi: 10.1109/MSEC.2019.2935666.

[6] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, "A Survey of Deep Learning Methods for Cyber Security," *Information 2019, Vol. 10, Page 122*, vol. 10, no. 4, p. 122, Apr. 2019, doi: 10.3390/INFO10040122.

[7] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 2, pp. 1153–1176, Apr. 2016, doi: 10.1109/COMST.2015.2494502.

[8] J. M. Torres, … C. I. C.-I. J. of, and undefined 2019, "Machine learning techniques applied to cybersecurity," *Springer*, Accessed: Mar. 15, 2023. [Online]. Available: https://link.springer.com/article/10.1007/s13042-018-00906-1

[9] T. Nguyen, & G. A.-I. communications surveys, and undefined 2008, "A survey of techniques for internet traffic classification using machine learning," *ieeexplore.ieee.org*, vol. 10, no. 4, pp. 56–76, Dec. 2008, doi: 10.1109/SURV.2008.080406.

[10] A. Buczak, E. G.-I. C. surveys & tutorials, and undefined 2015, "A survey of data mining and machine learning methods for cyber security intrusion detection," *ieeexplore.ieee.org*, Accessed: Mar. 15, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7307098/

[11] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," *Applied Soft Computing Journal*, vol. 10, no. 1, pp. 1–35, Jan. 2010, doi: 10.1016/J.ASOC.2009.06.019.

[12] J. Martínez Torres, C. Iglesias Comesaña, and P. J. García-Nieto, "Review: machine learning techniques applied to cybersecurity," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 10, pp. 2823–2836, Oct. 2019, doi: 10.1007/S13042-018-00906-1.

[13] S. Wu, W. B.-A. soft computing, and undefined 2010, "The use of computational intelligence in intrusion detection systems: A review," *Elsevier*, 2008, Accessed: Mar. 15, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494609000908

[14] D. Usynin, D. Rueckert, and G. Kaissis, "Beyond Gradients: Exploiting Adversarial Priors in Model Inversion Attacks," *Proceedings on Privacy Enhancing Technologies*, pp. 1–18.

[15] M. Malekzadeh, A. Borovykh, and D. Gündüz, "Honest-but-Curious Nets: Sensitive Attributes of Private Inputs Can Be Secretly Coded into the Classifiers' Outputs; Honest-but-Curious Nets: Sensitive Attributes of Private Inputs Can Be Secretly Coded into the Classifiers' Outputs", Accessed: Mar. 15, 2023. [Online]. Available: https://github.com/mmalekzadeh/honest-but-curious-nets.

[16] A. Moradi, N. K. D. Venkategowda, S. Pouria Talebi, and S. Werner, "Distributed Kalman Filtering with Privacy against Honest-but-Curious Adversaries".

[17] P. Mohassel and Y. Zhang, "SecureML: A System for Scalable Privacy-Preserving Machine Learning," *Proc IEEE Symp Secur Priv*, pp. 19–38, Jun. 2017, doi: 10.1109/SP.2017.12.

[18] E. Boyle, N. Gilboa, and Y. Ishai, "Secure Computation with Preprocessing via Function Secret Sharing," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11891 LNCS, pp. 341–371, 2019, doi: 10.1007/978-3-030-36030-6_14/FIGURES/1.

[19] N. Carlini, M. Jagielski, and I. Mironov, "Cryptanalytic extraction of neural network models," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12172 LNCS, pp. 189–218, 2020, doi: 10.1007/978-3-030-56877-1_7.

[20] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl Based Syst*, vol. 216, p. 106775, Mar. 2021, doi: 10.1016/J.KNOSYS.2021.106775.

[21] Z. Yu, J. Hu, G. Min, Z. Wang, W. Miao, and S. Li, "Privacy-Preserving Federated Deep Learning for Cooperative Hierarchical Caching in Fog Computing," *IEEE Internet Things J*, vol. 9, no. 22, pp. 22246–22255, Nov. 2022, doi: 10.1109/JIOT.2021.3081480.

[22] P. Kairouz *et al.*, "Advances and Open Problems in Federated Learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, Jun. 2021, doi: 10.1561/2200000083.

[23] K. Bonawitz *et al.*, "Practical secure aggregation for privacy-preserving machine learning," *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 1175–1191, Oct. 2017, doi: 10.1145/3133956.3133982.

[24] J. Liu, M. Juuti, Y. Lu, and N. Asokan, "Oblivious neural network predictions via MiniONN transformations," *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 619–631, Oct. 2017, doi: 10.1145/3133956.3134056.

[25] B. Darvish *et al.*, "DeepSecure: Scalable provably-secure deep learning," *Proc Des Autom Conf*, vol. Part F137710, Jun. 2018, doi: 10.1145/3195970.3196023.

[26] M. Sadegh Riazi, E. M. Songhori, C. Weinert, T. Schneider, O. Tkachenko, and F. Koushanfar, "Chameleon: A hybrid secure computation framework for machine learning applications," *ASIACCS 2018 - Proceedings of the 2018 ACM Asia Conference on Computer and Communications Security*, vol. 18, pp. 707–721, May 2018, doi: 10.1145/3196494.3196522.

[27] L. Fan *et al.*, "Rethinking Privacy Preserving Deep Learning: How to Evaluate and Thwart Privacy Attacks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12500 LNCS, pp. 32–50, 2020, doi: 10.1007/978-3-030-63076-8_3/FIGURES/8.

[28] C. Rechberger and R. Walch, "Privacy-Preserving Machine Learning Using Cryptography," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13049 LNCS, pp. 109–129, 2022, doi: 10.1007/978-3-030-98795-4_6/COVER.

[29] B. Darvish Rouhani, M. Sadegh Riazi, and F. Koushanfar, "DeepSecure: Scalable Provably-Secure Deep Learning," *ArXiv*, p. arXiv:1705.08963, May 2017, doi: 10.48550/ARXIV.1705.08963.

[30] A. Thantharate, "FED6G: Federated Chameleon Learning for Network Slice Management in Beyond 5G Systems," *2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2022*, pp. 19–25, 2022, doi: 10.1109/IEMCON56893.2022.9946488.

[31] F. Boemer, R. Cammarota, D. Demmler, T. Schneider, and H. Yalame, "MP2ML: A mixed-protocol machine learning framework for private inference," *ACM International Conference Proceeding Series*, Aug. 2020, doi: 10.1145/3407023.3407045.

[32] W.-S. Choi, B. Reagen, G.-Y. Wei, and D. Brooks, "Impala: Low-Latency, Communication-Efficient Private Deep Learning Inference," May 2022, Accessed: Mar. 27, 2023. [Online]. Available: https://arxiv.org/abs/2205.06437v1

[33] V. S. Naresh and M. Thamarai, "Privacy-preserving data mining and machine learning in healthcare: Applications, challenges, and solutions," *Wiley Interdiscip Rev Data Min Knowl Discov*, vol. 13, no. 2, p. e1490, Mar. 2023, doi: 10.1002/WIDM.1490.

[34] Y. Cai, Q. Zhang, R. Ning, C. Xin, and H. Wu, "Hunter: HE-Friendly Structured Pruning for Efficient Privacy-Preserving Deep Learning," *ASIA CCS 2022 - Proceedings of the 2022 ACM Asia Conference on Computer and Communications Security*, pp. 931–945, May 2022, doi: 10.1145/3488932.3517401.

[35] C. Juvekar, M. Mtl, V. Vaikuntanathan, and A. Chandrakasan, "GAZELLE: A Low Latency Framework for Secure Neural Network Inference", Accessed: Mar. 27, 2023. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/juvekar

[36] Z. Zhou, Q. Fu, Q. Wei, and Q. Li, "LEGO: A hybrid toolkit for efficient 2PC-based privacy-preserving machine learning," *Comput Secur*, vol. 120, p. 102782, Sep. 2022, doi: 10.1016/J.COSE.2022.102782.

[37] M. Sadegh Riazi, M. Samragh, H. Chen, K. Laine, K. Lauter, and F. Koushanfar, "XONN: XNOR-based Oblivious Deep Neural Network Inference," *ArXiv*, p. arXiv:1902.07342, Feb. 2019, doi: 10.48550/ARXIV.1902.07342.

[38] Y. Hong *et al.*, "A Privacy-Preserving Distributed Machine Learning Protocol Based on Homomorphic Hash Authentication," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13787 LNCS, pp. 374–386, 2022, doi: 10.1007/978-3-031-23020-2_21/COVER.

[39] W. Du, M. Li, X. Yang, L. Wu, and T. Zhou, "VCFL: A verifiable and collusion attack resistant privacy preserving framework for cross-silo federated learning," *Pervasive Mob Comput*, vol. 86, p. 101697, Oct. 2022, doi: 10.1016/J.PMCJ.2022.101697.

[40] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," *Proceedings of the ACM Conference on Computer and Communications Security*, vol. 2015-October, pp. 1310–1321, Oct. 2015, doi: 10.1145/2810103.2813687.

[41] Y. Wang, S. Ma, Q. Chen, J. Zhuang, and D. Jiang, "A geodesic projection-based data fusion scheme for cooperative spectrum sensing," *Digit Signal Process*, p. 104006, Mar. 2023, doi: 10.1016/J.DSP.2023.104006.

[42] P. Adong, E. Bainomugisha, D. Okure, and R. Sserunjogi, "Applying machine learning for large scale field calibration of low-cost PM2.5 and PM10 air pollution sensors," *Applied AI Letters*, vol. 3, no. 3, p. e76, Sep. 2022, doi: 10.1002/AIL2.76.

[43] S. Kotwal, P. Rani, T. Arif, J. Manhas, and S. Sharma, "Automated Bacterial Classifications Using Machine Learning Based Computational Techniques: Architectures, Challenges and Open Research Issues," *Archives of Computational Methods in Engineering*, vol. 29, no. 4, pp. 2469–2490, Jun. 2022, doi: 10.1007/S11831-021-09660-0/TABLES/5.

[44] S. Shen, T. Zhu, D. Wu, W. Wang, and W. Zhou, "From distributed machine learning to federated learning: In the view of data privacy and security," *Concurr Comput*, vol. 34, no. 16, p. e6002, Jul. 2022, doi: 10.1002/CPE.6002.

[45] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep Models under the GAN: Information leakage from collaborative deep learning," *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 603–618, Oct. 2017, doi: 10.1145/3133956.3134012.

[46] S. Dave *et al.*, "Special Session: Towards an Agile Design Methodology for Efficient, Reliable, and Secure ML Systems," *Proceedings of the IEEE VLSI Test Symposium*, vol. 2022-April, 2022, doi: 10.1109/VTS52500.2021.9794253.

[47] Y. Li, X. Wei, Y. Li, Z. Dong, and M. Shahidehpour, "Detection of False Data Injection Attacks in Smart Grid: A Secure Federated Deep Learning Approach," *IEEE Trans Smart Grid*, vol. 13, no. 6, pp. 4862–4872, Nov. 2022, doi: 10.1109/TSG.2022.3204796.

[48] G. Xu *et al.*, "SIMC 2.0: Improved Secure ML Inference Against Malicious Clients," Jul. 2022, doi: 10.48550/arxiv.2207.04637.

[49] B. Noordijk *et al.*, "baseLess: lightweight detection of sequences in raw MinION data," *Bioinformatics Advances*, vol. 3, no. 1, Jan. 2023, doi: 10.1093/BIOADV/VBAD017.

[50] A. Senanayake, H. Gamaarachchi, D. Herath, and R. Ragel, "DeepSelectNet: deep neural network based selective sequencing for oxford nanopore sequencing," *BMC Bioinformatics*, vol. 24, no. 1, pp. 1–16, Jan. 2023, doi: 10.1186/S12859-023-05151-0/FIGURES/7.

[51] M. S. Khan, B. Farzaneh, N. Shahriar, N. Saha, and R. Boutaba, “SliceSecure: Impact and Detection of DoS/DDoS Attacks on 5G Network Slices,” *2022 IEEE Future Networks World Forum (FNWF)*, pp. 639–642, Oct. 2022, doi: 10.1109/FNWF55208.2022.00117.

[52] R. Patan and R. M. Parizi, “Performance Improvement of Blockchain-based IoT Applications using Deep Learning Techniques,” *2022 4th International Conference on Blockchain Computing and Applications, BCCA 2022*, pp. 151–158, 2022, doi: 10.1109/BCCA55292.2022.9922342.

[53] A. Chohra, P. Shirani, E. M. B. Karbab, and M. Debbabi, “Chameleon: Optimized feature selection using particle swarm optimization and ensemble methods for network anomaly detection,” *Comput Secur*, vol. 117, p. 102684, Jun. 2022, doi: 10.1016/J.COSE.2022.102684.

[54] H. C. Tanuwidjaja, R. Choi, and K. Kim, “A Survey on Deep Learning Techniques for Privacy-Preserving,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11806 LNCS, pp. 29–46, 2019, doi: 10.1007/978-3-030-30619-9_4/COVER.

[55] H. Tian *et al.*, “Sphinx: Enabling Privacy-Preserving Online Learning over the Cloud,” *Proc IEEE Symp Secur Priv*, vol. 2022-May, pp. 2487–2501, 2022, doi: 10.1109/SP46214.2022.9833648.

[56] A. Boulemtafes, A. Derhab, and Y. Challal, “A review of privacy-preserving techniques for deep learning,” *Neurocomputing*, vol. 384, pp. 21–45, Apr. 2020, doi: 10.1016/j.neucom.2019.11.041.

[57] M. Sadegh Riazi *et al.*, “Xonn: XNOR-based Oblivious Deep Neural Network Inference”, Accessed: Mar. 16, 2023. [Online]. Available: https://www.usenix.org/conference/usenixsecurity19/presentation/riazi

[58] M. S. Riazi, B. Darvish Rouani, and F. Koushanfar, “Deep Learning on Private Data,” *IEEE Secur Priv*, vol. 17, no. 6, pp. 54–63, Nov. 2019, doi: 10.1109/MSEC.2019.2935666.

[59] K. Murdock, D. Oswald, F. D. Garcia, J. Van Bulck, D. Gruss, and F. Piessens, “Plundervolt: Software-based Fault Injection Attacks against Intel SGX,” 2020, doi: 10.1109/SP40000.2020.00057.

[60] P. Yuhala *et al.*, “Montsalvat: Intel SGX shielding for GraalVM native images Montsalvat: Intel SGX Shielding for GraalVM Native Images CCS CONCEPTS,” *22nd International Middleware Conference (Middleware ’21), December 6â•fi10, 2021, Virtual Event, Canada*, vol. 1, no. 2, pp. 352–364, 2021, doi: 10.1145/3464298.3493406ï.

[61] K. Bonawitz *et al.*, “Practical secure aggregation for privacy-preserving machine learning,” *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 1175–1191, Oct. 2017, doi: 10.1145/3133956.3133982.

[62] G. Lin, N. Sun, S. Nepal, J. Zhang, Y. Xiang, and H. Hassan, “Statistical Twitter Spam Detection Demystified: Performance, Stability and Scalability,” *IEEE Access*, vol. 5, pp. 11142–11154, 2017, doi: 10.1109/ACCESS.2017.2710540.

[63] J. Chen, W. H. Wang, and X. Shi, "Differential Privacy Protection Against Membership Inference Attack on Machine Learning for Genomic Data," in *Biocomputing 2021*, WORLD SCIENTIFIC, Nov. 2020, pp. 26–37. doi: 10.1142/9789811232701_0003.

[64] S. Guo, T. Zhang, G. Xu, H. Yu, T. Xiang, and Y. Liu, "Topology-Aware Differential Privacy for Decentralized Image Classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 4016–4027, Jun. 2022, doi: 10.1109/TCSVT.2021.3105723.

[65] S. Butt, M. Tariq, T. Jamal, A. Ali, … J. M.-I., and undefined 2019, "Predictive variables for agile development merging cloud computing services," *ieeexplore.ieee.org*, Accessed: Mar. 16, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8765563/

[66] M. T.-K. T. on I. and I. Systems and undefined 2019, "Agent based information security framework for hybrid cloud computing," *koreascience.or.kr*, Accessed: Mar. 16, 2023. [Online]. Available: https://www.koreascience.or.kr/article/JAKO201912261948438.page

[67] N. Carlini, D. W.-2017 ieee symposium on security and, and undefined 2017, "Towards evaluating the robustness of neural networks," *ieeexplore.ieee.org*, Accessed: Mar. 16, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7958570/

[68] … M. T.-J. of F. G. C. and and undefined 2018, "Analysis of the effectiveness of cloud control matrix for hybrid cloud computing," *researchgate.net*, vol. 11, no. 4, pp. 1–10, 2018, doi: 10.14257/ijfgcn.2018.11.4.01.

[69] X. Song, L. Wang, and X. Luo, "Airfoil optimization using a machine learning-based optimization algorithm," *J Phys Conf Ser*, vol. 2217, no. 1, p. 012009, Apr. 2022, doi: 10.1088/1742-6596/2217/1/012009.

[70] C. Szegedy *et al.*, "Intriguing properties of neural networks," *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, Dec. 2013, doi: 10.48550/arxiv.1312.6199.

[71] W. Li *et al.*, "Spear and Shield: Attack and Detection for CNN-Based High Spatial Resolution Remote Sensing Images Identification," *IEEE Access*, vol. 7, pp. 94583–94592, 2019, doi: 10.1109/ACCESS.2019.2927376.

[72] H. Hirano, A. Minagi, and K. Takemoto, "Universal adversarial attacks on deep neural networks for medical image classification," *BMC Med Imaging*, vol. 21, no. 1, pp. 1–13, Dec. 2021, doi: 10.1186/S12880-020-00530-Y/FIGURES/6.

[73] 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, doi: 10.1109/ACCESS.2018.2807385.

[74] P. Tabacof, J. Tavares, and E. Valle, "Adversarial Images for Variational Autoencoders," Dec. 2016, Accessed: Mar. 20, 2023. [Online]. Available: http://arxiv.org/abs/1612.00155

[75] N. Akhtar and A. Mian, "Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey".

[76] N. Papernot, P. McDaniel, … A. S.-M. 2016-2016, and undefined 2016, "Crafting adversarial input sequences for recurrent neural networks," *ieeexplore.ieee.org*, Accessed: Mar. 20, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7795300/

[77] Y. Song, T. Kim, S. Nowozin, … S. E. preprint arXiv, and undefined 2017, "Pixeldefend: Leveraging generative models to understand and defend against adversarial examples," *arxiv.org*, Accessed: Mar. 20, 2023. [Online]. Available: https://arxiv.org/abs/1710.10766

[78] D. Rumelhart, G. Hinton, R. W.- nature, and undefined 1986, "Learning representations by back-propagating errors," *nature.com*, Accessed: Mar. 20, 2023. [Online]. Available: https://www.nature.com/articles/323533a0

[79] Y. Song, T. Kim, S. Nowozin, … S. E. preprint arXiv, and undefined 2017, "Pixeldefend: Leveraging generative models to understand and defend against adversarial examples," *arxiv.org*, Accessed: Mar. 20, 2023. [Online]. Available: https://arxiv.org/abs/1710.10766

[80] X. Pan *et al.*, "Characterizing attacks on deep reinforcement learning," *arxiv.org*, Accessed: Mar. 20, 2023. [Online]. Available: https://arxiv.org/abs/1907.09470

[81] J. Sun *et al.*, "Stealthy and efficient adversarial attacks against deep reinforcement learning," *ojs.aaai.org*, Accessed: Mar. 20, 2023. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/6047

[82] S. Cho, T. J. Jun, B. Oh, and D. Kim, "DAPAS : Denoising Autoencoder to Prevent Adversarial attack in Semantic Segmentation".

[83] Z. Wang, B. Wang, Y. Liu, and J. Guo, "Global Feature Attention Network: Addressing the Threat of Adversarial Attack for Aerial Image Semantic Segmentation," *Remote Sensing 2023, Vol. 15, Page 1325*, vol. 15, no. 5, p. 1325, Feb. 2023, doi: 10.3390/RS15051325.

[84] Y. M. Khedr, Y. Xiong, and K. He, "Semantic Adversarial Attacks on Face Recognition through Significant Attributes," Jan. 2023, Accessed: Mar. 20, 2023. [Online]. Available: http://arxiv.org/abs/2301.12046

[85] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial Attacks and Defenses in Deep Learning," *Engineering*, vol. 6, no. 3, pp. 346–360, Mar. 2020, doi: 10.1016/J.ENG.2019.12.012.

[86] Y. Chen, "Celestial image classification based on deep learning and FGSM attack," *https://doi.org/10.1117/12.2656011*, vol. 12509, pp. 671–676, Jan. 2023, doi: 10.1117/12.2656011.

[87] S. Agnihotri and M. Keuper, "CosPGD: a unified white-box adversarial attack for pixel-wise prediction tasks," Feb. 2023, Accessed: Mar. 22, 2023. [Online]. Available: https://arxiv.org/abs/2302.02213v1

[88] A. Raghunathan, J. Steinhardt, and P. Liang, "Certified defenses against adversarial examples," *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 2018.

[89] A. Raghunathan, … J. S.-A. in neural, and undefined 2018, "Semidefinite relaxations for certifying robustness to adversarial examples," *proceedings.neurips.cc*, Accessed: Mar. 23, 2023. [Online]. Available:

https://proceedings.neurips.cc/paper/2018/hash/29c0605a3bab4229e46723f89cf59d83-Abstract.html

[90] Y. Guo, C. Zhang, C. Zhang, and Y. Chen, "Sparse dnns with improved adversarial robustness," *proceedings.neurips.cc*, Accessed: Mar. 23, 2023. [Online]. Available: https://proceedings.neurips.cc/paper/2018/hash/4c5bde74a8f110656874902f07378009-Abstract.html

[91] K. Xiao, V. Tjeng, N. Shafiullah, A. M. preprint arXiv, and undefined 2018, "Training for faster adversarial robustness verification via inducing relu stability," *arxiv.org*, Accessed: Mar. 23, 2023. [Online]. Available: https://arxiv.org/abs/1809.03008

[92] T.-W. Weng *et al.*, "Evaluating the robustness of neural networks: An extreme value theory approach," *arxiv.org*, Accessed: Mar. 23, 2023. [Online]. Available: https://arxiv.org/abs/1801.10578

[93] C. Sitawarin and D. Wagner, "Minimum-Norm Adversarial Examples on KNN and KNN based Models," *Proceedings - 2020 IEEE Symposium on Security and Privacy Workshops, SPW 2020*, pp. 34–40, May 2020, doi: 10.1109/SPW50608.2020.00023.

[94] C. Sitawarin and D. Wagner, "On the robustness of deep K-nearest neighbors," *Proceedings - 2019 IEEE Symposium on Security and Privacy Workshops, SPW 2019*, pp. 1–7, May 2019, doi: 10.1109/SPW.2019.00014.

[95] Y.-Y. Yang, "Adversarial Examples for Non-Parametric Methods: Attacks, Defenses and Large Sample Limits.," *CoRR*, vol. abs/1906.03310, 2019, Accessed: Mar. 23, 2023. [Online]. Available: http://arxiv.org/abs/1906.03310

[96] H. Sasahara and H. Sandberg, "Asymptotic Security using Bayesian Defense Mechanism with Application to Cyber Deception," *IEEE JOURNAL*, vol. XX, Jan. 2022, Accessed: Mar. 23, 2023. [Online]. Available: https://arxiv.org/abs/2201.02351v2

[97] M. Yin, S. Li, C. Song, M. S. Asif, A. K. Roy-Chowdhury, and S. V. Krishnamurthy, "ADC: Adversarial Attacks Against Object Detection That Evade Context Consistency Checks." pp. 3278–3287, 2022.

[98] Y. Man, R. Muller, M. Li, Z. Berkay Celik, and R. Gerdes, "That Person Moves Like A Car: Misclassification Attack Detection for Autonomous Systems Using Spatiotemporal Consistency".

[99] L. Zhang, Y. Pan, Y. Liu, Q. Zheng, and Z. Pan, "Multiple Domain Cyberspace Attack and Defense Game Based on Reward Randomization Reinforcement Learning," *XXXX*, vol. 16, no. 7, p. 1, May 2022, Accessed: Mar. 23, 2023. [Online]. Available: https://arxiv.org/abs/2205.10990v1