International Interdisciplinary Virtual Conference on
'Recent Advancements in Computer Science, Management and Information Technology'
International Journal of Scientific Research in Computer Science,
Engineering and Information Technology| ISSN : 2456-3307 (www.ijsrcseit.com)

A Survey on Defensive Measures to Secure Machine Learning Systems

Akshay Dilip Lahe[1], Dr. Guddi Singh[2]
[1]Research Scholar, [2]Research Supervisor
Kalinga University, Raipur, Chhattisgarh, India

ABSTRACT

In recent years, Machine learning is being used in various systems in wide variety of applications like Healthcare, Image processing, Computer Vision, Classifications, etc. Machine learning have shown that it can solve complex problem easily more efficiently than human beings. But through wide research it is found that security of ML systems can be compromise by various attacks. This survey aims to analyse various defence mechanisms and measures which can protect the complete machine learning pipeline against various attacks. We are categorizing them depending on position of attacks in machine learning pipeline. This paper will focus on all aspects of ML security at various stages from training phase to testing phase instead of focusing on one type of security countermeasure.

Keywords— Artificial Intelligence Security, Machine Learning Security, Poisoning attacks, backdoor attacks, adversarial attacks, Security Attacks in ML

I. INTRODUCTION

Machine Learning that comes under computational algorithm used to mimic human learning and decision capacities deduced from its environment are being widely used in various domains like computer vision, engineering, banking and finance, entertainment industry, smart mobile and web applications, biomedical and healthcare applications. With increase of accumulation of huge amount of data and with emergence of concept of big data, various data mining and machine learning techniques have been developed for pattern recognition, future predictions, decision making along with other application tasks. Machine learning is based on concept of mimicking human beings' way of learning things along with sensory input processing to achieve a particular task.

Machine Learning (ML) can be described as ability to learn without programmed explicitly. ML algorithms learn how to perform certain task based on input data given to the algorithm and perform same task when presented with new data. ML model is trained on training data which include multitude of features which is called as Learning Phase. Then ML model is tested by presenting new data to the model and it should give correct result as per learning phase. This phase is called as Testing Phase. Using metrics like accuracy to predict correct result as per learning, and precision, performance of ML model is measured. The accuracy can depend

on factors like quantity of training data, ML Algorithm used, feature selection, feature extraction method used and hyper parameters.

This survey focuses on following points:

1. ML models and various defences that can be used at various positions of ML pipeline instead of just focusing on one stage.
2. Defence mechanisms are divided based on location as well as training or testing phase and categorize them in a well formatted manner.

II. APPLICATIONS OF MACHINE LEARNING

In recent decade, research in Machine Learning algorithms and its models have drastically increased and various approaches has been proposed. As of now, ML is being used in almost all the domains which include computer vision, prediction, market analysis, semantic analysis, NLP, healthcare, Information management systems, Network security, medical diagnosis and Healthcare sectors [1].

Object detection and recognition and its processing are using ML/DL in computer vision domain. For application which operates for prediction are using ML for classification purpose of documents, images and faces. Image analysis and segmentation is used for medical diagnosis. For security of various systems, ML is being used in IDS and for anomaly detection along with network intrusion and privacy aware systems to provide security to various applications. DoS attacks can be predicted using machine learning approaches. ML is used commonly in semantic analysis, NLP and information retrieval. K-NN and SVM are used to recognize hand gestures. Text classification can be done using linear classification, ANN and SVM effectively. Recommender systems have been built using ML in both bioinformatics and mobile advertisement domain. In Network Security, ML is used for IDPS, Endpoint protection which include malware classification and detection, access control and authentication detection. Process anomaly detection and fraud detection can be done by processing behaviors using ML models. User behavior can be observed using ML models which include keystroke dynamics detection and breaking human interaction proofs. ML can provide security to application by providing detections of malicious URL, phishing and spam [2].

ML in Healthcare is recent emerging domain of ML application. Large data is being generated by healthcare information systems with the introduction of electronic health records so it becomes complex to analyze, process and mine useful information using traditional methods. ML helps to analyze this data and provide insights to doctors or other stakeholders in healthcare. Prognosis meaning predicting expected future outcomes of the disease in clinical environment can be done using ML models. ML can be used for diagnosis purposes by analyzing EHRs on regular basis of the patients. Healthcare domain uses MRI, CT, Ultrasound scans to diagnose diseases. Image analysis along with ML can be used on these scans to effectively diagnose a disease. Extensive research is being conducted on application of ML which monitors patient's health in continuous manner with the help of wearable devices having sensors [3].

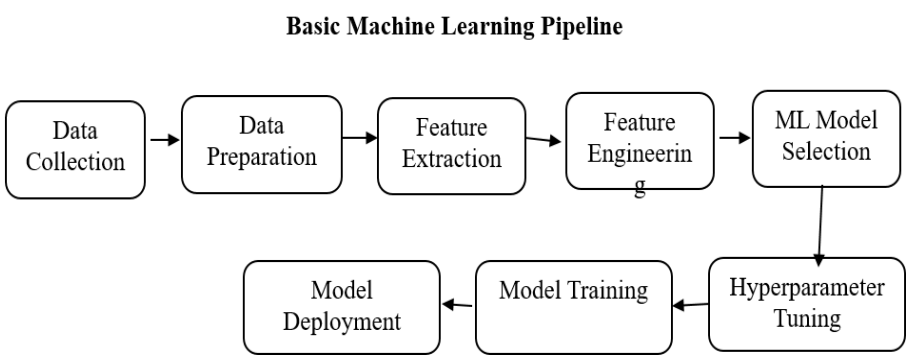

Fig. 1: Basic Stages of ML model

Fig.1 shows various stages of generic ML model. First data relevant to application is collected at one place. This collected data can be dirty or noisy so it needs some preparations and cleaning before giving it to the ML model. Data preparation stage cleans the data, pre-process it and prepare it for the feature extraction. In feature extraction stage, important or significant features are selected and extracted out of the prepared data. Features which impact the model's outcome are selected here. Then addition and removal of various features or creation of artificial features can be done in feature engineering stage. There are various types of ML models so based on the application and input features, best approach model is selected and trained on input data. To increase the performance of ML model, we can adjust the hyper-parameters. After successful training and then testing of model on new data, it is deployed in the real-world application.

III. SECURITY OF ML SYSTEMS

Despite its application in wide domains, research in security of Machine Learning Models is comparatively less. There are different components of a general ML Model which includes Raw data, Datasets (training, validation and test), Learning algorithms, Evaluation methods, ML model itself, Output of the model, etc. All of these components are prone to risks from an attacker. ML model can be at a risk from attackers and there are various types of attacks which can be performed [3].

Broadly, Machine learning system has two major stages:

1. **Training Stage:** training input data is fed to model as input and model is trained using this data using some algorithm.
2. **Testing Stage:** train ed model is presented with new data called test data to see if model is performing as per expectations.

Attacker can attack at both of these major stages to gain sensitive information. Machine Learning system is vulnerable to various stages and points throughout its pipeline. Data poisoning and backdoor attacks can be done on training data to misled the model. Model's output can be compromised by model theft and recovery of training data from model's output. Various adversarial attacks can craft to misled the model output the test input data. Through these examples, it is evident that ML model itself are vulnerable at many points. In this research survey, I am to give a details category wise classification of these attacks and vulnerabilities of ML pipeline.

IV. GOAL OF ATTACKER

Goal or Aim of an Attacker can be presented in three aspects i.e., Violation of Security, Specificity of attack and Influence of Attacks [4] [5].

Violation of Security: There are three major violations that an attacker can cause: Integrity violation in which intrusive points can be classified as normal to avoid detection without compromising system functionalities; Availability violation in which attacker causes so many false errors that system functionality becomes unavailable to legitimate users of the systems; Privacy violation comprises leaking of sensitive private information to attacker.

Specificity of Attack: An attack can be a targeted attack which focuses to cause harm to a set of samples or points or it can be indiscriminate attack which is more flexible attack focusing on a general class of samples or any sample.

Influence of Attack: It can be of two types: Causative attack which influence training data of model and alter the training process; Exploratory attack discover information about training data using techniques like probing.

V. VULNERABILITIES AT VARIOUS STAGES OF MACHINE LEARNING SYSTEM IN HEALTHCARE DOMAIN

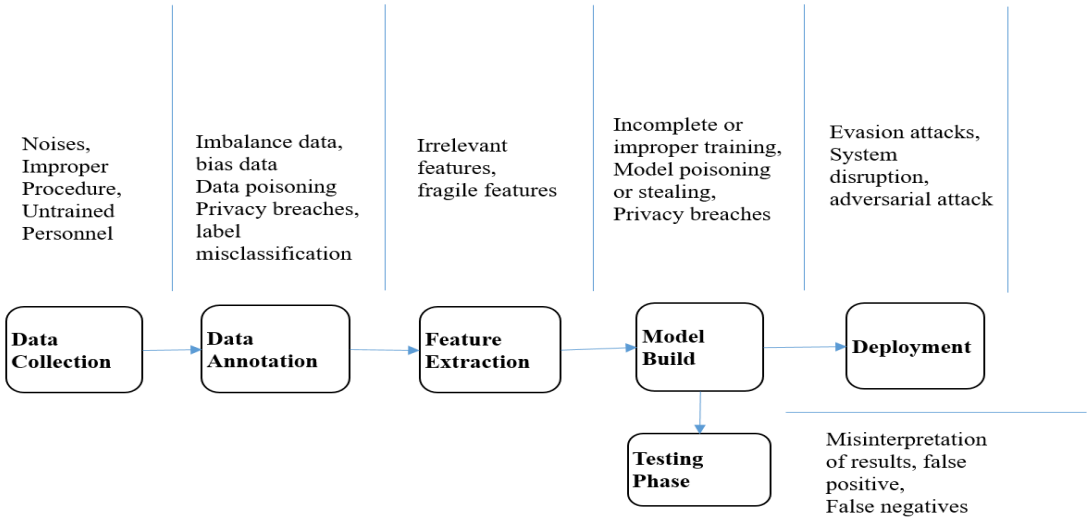

Fig. 2: Healthcare Machine Learning Pipeline and Vulnerabilities at various stages

The fig. 2 shows example of various vulnerabilities at different stages of machine learning pipeline in healthcare domain. This will illustrate how a ML model is vulnerable at various stages. At Data collection stage, there are vulnerabilities which include noises, dirty data, missing data, improper procedure, untrained personnel, etc. Imbalance data, biased data, data poisoning, privacy breaches, label misclassification and label leakages are few vulnerabilities at data annotation stage of ML system in healthcare [2]. During feature extraction, there can be fragile features or irrelevant features and knowledge of feature selection algorithms or features set can help attacker. While training the model using training input data, input data is vulnerable to data poisoning attack or there can be backdoor which can help an attacker to gain access to the model. Model poisoning or stealing attack can cause model to misclassify. Evasion attacks, system disruption, network issues, adversarial attacks can be done at test data or model's output.

As given in our example ML Pipeline, ML attacks can happen at various phases of ML lifecycle. Here we are categorizing attacks into two main categories: 1. Attacks during training phase and 2. Attacks during Testing Phase and Model's Output.

1. Attacks during Training Phase
   - A. Poisoning training data
   - B. Backdoor in Training data
2. Attacks during Testing Phase and Model's Output
   - A. Adversarial Attacks
     - i. Having Knowledge of system
     - ii. Without any knowledge of system
   - B. Model Extraction Attack
   - C. Stealing Hyper Parameters
   - D. Training input data recovery
     - i. Model Inversion
     - ii. Data Membership

VI. DEFENSIVE MEACHNISMS FOR ML-PIPELINE

This section refers to various countermeasures proposed to secure the machine learning pipeline at various stages against various attacks. This paper categorizes the defenses in five sub-sections which are analyzed and summarized in this section. The categorization is as follow

1. Poisoning Attack Defense
2. Backdoor Attack Defense
3. Adversarial Attack Defense
4. Model Stealing Attack Defense
5. Sensitive Data Protection

1. Poisoning Attack Defense

An attacker can manipulate the training input data or inject malicious data as training input data which will compromise ML model. Poisoning attacks can lead to serious consequences, such as biased predictions, degraded accuracy, and compromised security.

Machine learning systems have become ubiquitous in many domains like healthcare, computer vision, speech, etc. These systems rely on training input data to learn patterns and make predictions. However, Attacker can manipulate this input data to lower performance of the system or to confuse the system or to make system bias as per attacker's desired output. Poisoning attacks can be targeted or untargeted and can lead to serious consequences, such as biased predictions, degraded accuracy, and compromised security.

We divide this into three categories: data-centric, model-centric, and hybrid approaches.

1.1 Data-Centric Methods:

Data-centric methods detect and remove the malicious data from input training data of the model. These approaches [7] [8] [9] include:

Data sanitization: This approach involves preprocessing the training data to remove or neutralize malicious samples. Various techniques have been proposed, such as clustering-based methods, outlier detection, and density-based methods.

Data diversity: This approach involves creating training input data which have huge diversity among the data. This can reduce the effect of malicious data injection. Techniques such as ensemble learning, data augmentation, and adversarial training fall into this category.

1.2 Model-Centric Methods:

Model-centric methods focus on designing robust models that can resist poisoning attacks. These approaches include:

Robust optimization: This approach involves creating ML model such that there will be huge gap between defender and attacker. Robustness of the model system is increased in this method. Various techniques such as adversarial training, regularization, and optimization-based methods fall into this category.

Model selection: This approach involves selecting models that are less susceptible to poisoning attacks. Techniques such as robust decision trees, random forests, and gradient boosting fall into this category.

1.3 Hybrid Methods:

These methods are combination of first and second methods which will provide a more comprehensive defense against poisoning attacks. These approaches include:

Ensemble learning: In this method, we train different ML models on different input training data and then combine these models. Different methods like stacking, boosting, bagging, etc. comes under this category.

Defense distillation: This approach involves training a robust model that can detect poisoned data and using it to distill a simpler model that is more efficient and less susceptible to attacks.

ANTIDOTE is a technique proposed by Rubinstein et al. [10] which uses an anomaly detector to defend against poisoning attack by finding outliers in turn improving robustness of the ML model. But this approach mainly focuses on binary classification. Another Approach by Biggio et al. [11] uses bagging classifiers and ensemble method to reduce outliers and to secure against poisoning attack. There are two applications namely, Spam filtering and web-based IDS which have used this approach to defend against poisoning attack.

Game Theory can be used to create a SVM which analyses conflicts between attacker and learner. Zhang and Ahu [12] have created distributed SVM which predicts outcome of learner which in turn prevent updates which are wrong and prevent data poisoning. But this approach is expensive in terms of computation power.

Robust linear regression method given by Liu et al. [13] uses matrix factorization (low-rank) and then PCA Regression which prunes poisoned samples. TRIM is a defensive mechanism presented by Jagielski et al. [14] estimate regression parameters iteratively then remove and isolate doubtful poisonous samples with the use of trimmed loss function.

Open Research Challenges:

Despite the progress in defending against poisoning attacks, several open research challenges remain. These challenges include:

Adversary-aware defenses: Current defenses assume that the attacker follows a specific strategy, which may not be realistic in practice. Therefore, developing defenses that can handle unknown or adaptive attacks is an important research direction.

Scalability and efficiency: Many defenses are computationally expensive or require a large amount of memory. So there is need to develop efficient and scalable defenses.

2. Backdoor Attack Defense

One of the biggest attacks performed on ML model is backdoor attack. These attacks compromise the security and integrity of the system. It involves the insertion of a hidden trigger into a machine learning model during the training phase. This trigger can then be activated during the inference phase to cause the model to output incorrect results or perform malicious actions. Therefore, it is crucial to develop effective defenses against backdoor attacks. We categorize these defenses into four groups: training data, model architecture, detection & prevention.

Many defense mechanisms are there which addresses ML backdoor attack. These defenses can be classified into four groups: training data defenses, model architecture defenses, detection defenses, and prevention defenses. Training data defenses involve modifying the training data to prevent backdoor attacks. Model architecture defenses involve designing the machine learning model to be resistant to backdoor attacks. Detection defenses involve detecting backdoor attacks during the inference phase. Prevention defenses involve preventing backdoor attacks from occurring in the first place.

We identified several defenses [15][16] against backdoor attacks in machine learning systems. Training data defenses include techniques such as data sanitization, data augmentation, and outlier detection. Model architecture defenses include methods such as randomized smoothing, feature squeezing, and defensive distillation. Detection defenses include approaches such as input reconstruction, model interpretation, and output comparison. Prevention defenses include techniques such as adversarial training, regularization, and anomaly detection.

Chen et al. [17] gave an approach which can detect poisonous training input data along with activation clustering technique which can be performed under various backdoor scenarios. Liu et al. [18] use 3 approaches to protect model against trojans. This model requires more computational overhead and can be used for Deep Learning systems. STRIP is a run time trojan detection method proposed by Gao et al. [19]

While there has been significant progress in defenses against backdoor attacks, these defenses are not perfect and can have limitations. For example, training data defenses may not be effective against attacks that are specifically designed to evade such defenses. Model architecture defenses may not be practical for certain types of machine learning models or may require additional computational resources. Detection defenses may have high false positive rates, leading to unnecessary interventions.

3. Adversarial Attack Defense

These attacks are very popular and one of the major attacks that is performed on ML model. These attacks can manipulate behavior of ML model which leads to incorrect or malicious outputs. In recent years, researchers have proposed various defenses against adversarial attacks.

Adversarial attacks can have serious consequences, such as misinterpretation of road sign or incorrect medical diagnosis. Therefore, it is important to develop effective countermeasures that will provide protection from these attacks.

There are many defense methods presented in [20] which provide protection against adversarial attacks. One category of defenses is based on adversarial training, where adversarial examples are generated while training

and then model is trained using these examples. Other defenses include input preprocessing techniques, such as denoising and feature squeezing, and post-processing techniques, such as ensemble methods and detection-based approaches. In addition, there are defenses based on model verification, such as randomized smoothing and certified defenses.

We identified several white box and black box defenses against adversarial attacks. In white box approach attacker knows the model completely and on the other hand, in black-box approach attacker has limited or no information of the model. Adversarial training is a widely used white-box defense, which involves training the model using adversarial examples generated during training. Other white-box defenses include gradient masking, where the model's gradients are modified to prevent adversarial perturbations, and defensive distillation, where the model is trained using softened outputs from another model.

Black-box defenses include input preprocessing techniques, such as feature squeezing and input transformations, and detection-based approaches, such as outlier detection and anomaly detection. Model verification techniques, such as randomized smoothing and certified defenses, can also be used as black-box defenses.

Earlier in Models which do not use neural network approaches like game theory [21], feature detection [22], malicious pdf detection [23] and SecureDroid [24] were used to defend the system against adversarial attacks.

For neural network-based model, we can further categorize algorithms in two types i.e., complete defense technique and detection only technique [25]. First identified correct label whereas second identifies if input instance is adversarial or not. There are various complete defense approaches. Goodfellow et al. [26] gave defense called FGSM which is based on adversarial training, Saddle point formulation method [27], input transformations approach [28] and MagNet [29] which detects adversarial example using detector network.

While there have been significant advances in defenses against adversarial attacks, these defenses are not foolproof and can have limitations. For example, adversarial training can only mitigate attacks within a certain distance from the original input, and detection-based approaches may not work well for attacks with small perturbations. In addition, some defenses may be computationally expensive or require additional training data.

4. Model Stealing Attack Defense

In model stealing attack, attacker attempts to steal a machine learning model trained by another entity which compromise security, privacy and owner's rights of the model. Therefore, it is important to develop effective defenses against this type of attack. In recent years, researchers have proposed various defenses against model stealing.

Several defenses have been proposed [30][31] to mitigate model stealing attacks. One category of defenses is based on watermarking techniques, where a watermark is embedded into the model during training. The watermark can be used to detect if the model has been stolen. Other defenses include model obfuscation, where the model is modified to make it harder to understand; model splitting, where split the model and train separately and then combine output approach is followed.

We identified several theoretical and practical approaches to defend against model stealing attacks. Watermarking-based defenses include methods such as Digital Fingerprinting, Secret Watermarking, and Adversarial Watermarking. Model obfuscation defenses include techniques such as Deep Obfuscation, Model Pruning, and Function-Preserving Encryption. Model splitting defenses include methods such as Progressive Learning and Distributed Learning.

A Method which injects deceptive noises in confidence information which in turn mislead the attacker is proposed by Lee et. Al. [32]. If attacker send huge number of queries, then attack can succeed and this is the disadvantage of this method. Hanzlik et al. [33] presented MLCapsule framework which allows a model to be run on user's side in turn protecting user's own privacy and server will protect the intellectual property rights. In this method, encryption is required which becomes computational overhead.

While there have been significant advances in defenses against model stealing attacks, these defenses are not foolproof and can have limitations. In addition, some defenses may be computationally expensive or require additional training data.

5. Sensitive Data Protection

Machine learning is an essential tool for organizations and companies looking to utilize data to improve their products and services. However, this reliance on data comes with risks, particularly with regards to the privacy of sensitive training data. If this data falls into the wrong hands, it can be used for malicious purposes, leading to a significant loss of trust in the organization.

To address these concerns, privacy-protected machine learning techniques [34][35] have been developed to protect sensitive training data from recovery. we will discuss various techniques that can be used to protect sensitive data.

Differential Privacy

When presence or absence of any particular training data does not affect model's outputs then it is called as differential privacy technique. This is achieved by adding noise to the training data, ensuring that any data that is sensitive or identifiable cannot be recovered by malicious actors. Privacy budget parameter controls the noise amount that should be added, which determines the amount of information that can be leaked without compromising sensitive data's privacy.

One of the primary merits of this approach is that it can be applied to existing machine learning models without requiring any significant modifications. However, model's accuracy is reduced as a result of noise addition. Additionally, ensuring that the privacy budget is correctly set is essential to guarantee that the data remains secure.

Homomorphic Encryption

It encrypts the data while still allowing computations to be performed on it. This means that sensitive training data can be encrypted before being sent to a third-party for analysis, preventing any unauthorized access to the data. Additionally, homomorphic encryption also allows for the secure sharing of models, enabling multiple organizations to collaborate on machine learning projects without risking data breaches.

While homomorphic encryption can provide robust security to sensitive data, it comes with some significant drawbacks. Firstly, it requires significant computational power to perform encryption and decryption, leading to slower training times. Additionally, the use of homomorphic encryption can also reduce model's accuracy due to the need for additional computational resources.

Federated Learning

It allows multiple organizations to collaborate on machine learning projects while keeping their data private. This is achieved by each organization training a model on their data, and the models are then combined to create a final model. This approach ensures that sensitive data is kept secure while allowing multiple organizations to benefit from the insights generated by the machine learning model.

Federated learning has several advantages over other privacy-protected machine learning techniques. Firstly, it allows for the secure sharing of models without requiring any modifications to the underlying algorithms. Additionally, federated learning can lead to significant improvements in the accuracy of the model, as it enables the use of more diverse datasets.

There are various cryptography-based approaches which can protect sensitive data. Abadi et al. [36] presented differential privacy-based method which achieves privacy, complexity, model quality and efficiency. Phong et al. [37] proposed privacy preserving distributed learning framework but it can reveal sensitive data to server.

Another privacy preserving distributed framework is presented by Shokri and Shmatikov [38] where parallel stochastic gradient descent can be executed. Federated Learning framework-based multi-party computation is proposed by Bonawitz et al. [39] which performs a secure aggregation to provide security and privacy.

VII. RECOMMENDATIONS

1. ML models are susceptible to different attacks. The existing defense mechanisms are not sufficient to adhere security & privacy of ML systems.
2. This Survey will guide researcher through various security defense mechanisms to provide privacy and security to ML system at various stages of its pipeline.
3. There is need of robust and computationally fast privacy preserving defensive mechanisms for ML systems and this survey will help research to design such system.

VIII. FUTURE SCOPE

Research is being conducted on Machine Learning extensively. Following are some future directions regarding this:

- **Attack in real scenarios:** Most of attacks proposed are done in simulating environment and major part of the research of security of Machine Learning algorithms is done using simulating environment. But conditions and setups in real world can impact these algorithms and security attacks differently that is why there is need of conducting research in real world scenarios.
- **Security for ML Models:** Security of ML Models are to be focused more in research so as to provide a robust and secure ML Models which can directly be implemented in various applications.
- **Privacy preserving ML Models:** With increased attention to ML application, there is need of focus on privacy-aware or privacy-preserving architecture and approaches of Machine Learning Models. There are several privacies related issues like access control, protection of model's parameters from service providers, protecting sensitive information from third parties, etc. There is need of improving efficiency of cryptographic approaches in ML.

IX. REFERENCES

[1]. Adnan Qayyum, Junaid Qadir, Muhammad Bilal, Ala Al-Fuqaha, "Secure and Robust Machine Learning for Healthcare: A Survey", in 2020 IEEE Reviews in Biomedical Engineering (Volume: 14) DOI: 10.1109/RBME.2020.3013489.

[2]. Gary McGraw, Richie Bonett, Victor Shepardson, and Harold Figueroa, "The Top 10 Risks of Machine Learning Security" in IEEE: Computer (Volume: 53, Issue: 6, June 2020) DOI: 10.1109/MC.2020.2984868.

[3]. Marco Barreno, Blaine Nelson, Anthony D. Joseph, J.D. Tygar, "The security of machine learning" in Springer Machine Learning-volume 81 May 2010, page 121-148, DOI:10.1007/s10994-010-5188-5

[4]. Huang Xiao, Battista Biggio, Gavin Brown, Giorgio Fumera, Claudia Eckert, Fabio Roli, "Is feature selection secure against training data poisoning?" published in ICML 6 July 2015 Computer Science arXiv:1804.07933

[5]. P. Li, Q. Liu, W. Zhao, D. Wang, S. Wang, "Chronic poisoning against machine learning based IDSs using edge pattern detection," in IEEE International Conference on Communications (ICC 2018).

[6]. Biggio, B., Fumera, G., Roli, F., Didaci, L.,"Poisoning Adaptive Biometric System" in:, et al. Structural, Syntactic, and Statistical Pattern Recognition. SSPR /SPR 2012. Lecture Notes in Computer Science, vol 7626. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-34166-3_46.

[7]. Wang, T., Chen, M., Li, S., Huang, L., & Chen, Y. (2021). A survey on poisoning attacks and defenses in machine learning: Trends, challenges, and prospects. arXiv preprint arXiv:2102.09601.

[8]. Gao, J., Su, Y., Huang, X., & Zhang, H. (2020). A survey on poisoning attacks and defenses in machine learning. IEEE Access, 8, 200089-200106.

[9]. Munoz-Gonzalez, L., Biggio, B., Demontis, A., Paudice, A., Wongrassamee, V., & Lupu, E. (2017). Towards poisoning of deep learning algorithms with back-gradient optimization. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (pp. 27-38).

[10]. B. I. Rubinstein et al., "ANTIDOTE: Understanding and defending against poisoning of anomaly detectors," in Proc. 9th ACM SIGCOMM Int. Meas. Conf., Nov. 2009, pp. 1–14.

[11]. B. Biggio, I. Corona, G. Fumera, G. Giacinto, and F. Roli, "Bagging classifiers for fighting poisoning attacks in adversarial classification tasks," in Proc.10th Int. Conf. Mult. Classif. Syst., Jun. 2011, pp. 350–359.

[12]. R. Zhang and Q. Zhu, "A game-theoretic defense against data poisoning attacks in distributed support vector machines," in 56th IEEE Annu. Conf. Decis. Control, Dec. 2017, pp. 4582–4587.

[13]. C. Liu, B. Li, Y. Vorobeychik, and A. Oprea, "Robust linear regression against training data poisoning," in Proc. 10th ACMWorkshop Artif. Intel. Secur., Nov. 2017, pp. 91–102.

[14]. 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 IEEE Symp. Secur. Priv., May 2018, pp. 19–35.

[15]. Chen, X., Liu, C., Li, D., Song, D., & Wang, X. (2017). Detecting backdoor attacks on deep neural networks by activation clustering. arXiv preprint arXiv:1704.02654.

[16]. -Gu, T., Dolan-Gavitt, B., & Garg, S. (2020). Badnets: Identifying vulnerabilities in the machine learning model supply chain. In Proceedings of the 2020 USENIX Conference on Usenix Annual Technical Conference (pp. 109-124).

[17]. X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial examples: Attacksand defenses for deep learning," IEEE Trans. Neural Netw. Learn. Syst.,vol. 30, no. 9, pp. 2805–2824, Sep. 2019.

[18]. Y. Liu, Y. Xie, and A. Srivastava, "Neural Trojans," in IEEE Int. Conf. Comput. Des., Nov. 2017, pp. 45–48.

[19]. 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 Proc. 35th Annu. Comput. Secur. Appl. Conf., Dec. 2019, pp. 113–125

[20]. Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In Proceedings of the International Conference on Learning Representations.

[21]. N. Dalvi, P. Domingos, Mausam, S. Sanghai, and D. Verma, "Adversarial classification," in Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., Aug. 2004, pp. 99–108.

[22]. A. Globerson and S. Roweis, "Nightmare at test time: Robust learning by feature deletion," in Proc. 23rd Int. Conf. Mach. Learn., Jun. 2006, pp. 353–360.

[23]. N. Šrndi´c and P. Laskov, "Detection of malicious PDF files based on hierarchical document structure," in Proc. 20th Annu. Netw. Distrib. Syst. Secur. Symp., Feb. 2013, pp. 1–16.

[24]. L. Chen, S. Hou, and Y. Ye, "SecureDroid: Enhancing security of machine learning-based detection against adversarial android malware attacks," in Proc. 33rd Annu. Comput. Secur. Appl. Conf., 2017, pp. 362–372.

[25]. N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," IEEE Access, vol. 6, pp. 14 410–14 430, Jul. 2018.

[26]. I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in 3rd Int. Conf. Learn. Represent., May 2015, pp.1–11.

[27]. A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in 6th Int. Conf. Learn. Represent., May 2018, pp. 1–10.

[28]. C. Guo, M. Rana, M. Cissé, and L. van der Maaten, "Countering adversarial images using input transformations," in 6th Int. Conf. Learn. Represent., May 2018, pp. 1–12.

[29]. D. Meng and H. Chen, "MagNet: A two-pronged defense against adversarial examples," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur., Oct. 2017, pp. 135–147.

[30]. Song, X., Shu, T., Chen, X., Yang, Y., & Liu, Y. (2019). On the feasibility of embedding watermarking-based defense mechanisms in deep neural networks. In Proceedings of the International Joint Conference on Artificial Intelligence.

[31]. Wang, Y., Zeng, S., Cao, B., & Zhang, Z. (2021). Progressive learning for secure model aggregation. IEEE Transactions on Dependable and Secure Computing.

[32]. T. Lee, B. Edwards, I. Molloy, and D. Su, "Defending against neural network model stealing attacks using deceptive perturbations," in IEEE Secur. Priv. Workshops, May 2019, pp. 43–49.

[33]. L. Hanzlik et al., "MLCapsule: Guarded offline deployment of machine learning as a service," arXiv:1808.00590, 2018. [Online]. Available: http://arxiv.org/abs/1808.00590

[34]. Aono, Y., Duchi, J. C., Jordan, M. I., & Li, Z. (2019). Privacy-preserving decentralized optimization using local and shared gradients. arXiv preprint arXiv:1909.06926.

[35]. Shokri, R., & Shmatikov, V. (2015). Privacy-preserving deep learning. In Proceedings of the 22nd ACM SIGSAC conference on computer and communications security (pp. 1310-1321).

[36]. M. Abadi et al., "Deep learning with differential privacy," in Proc. ACM SIGSAC Conf. Comput. Commun. Secur., Oct. 2016, pp. 308–318.

[37]. L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacypreserving deep learning via additively homomorphic encryption," IEEE Trans. Inf. Forensics Secur., vol. 13, no. 5, pp. 1333–1345, May 2018.

[38]. R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in Proc.22nd ACM SIGSAC Conf. Comput. Commun. Secur., Oct. 2015, pp. 1310–1321.

[39]. K. Bonawitz et al., "Practical secure aggregation for federated learning on user-held data," arXiv:1611.04482, 2016. [Online]. Available: http://arxiv.org/abs/1611.044