2022 7th IEEE International Conference on Data Science in Cyberspace (DSC) | 978-1-6654-7480-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/DSC55868.2022.00014

A Survey on Data Poisoning Attacks and Defenses

Jiaxin Fan[1], Qi Yan[1], Mohan Li[1,2], Guanqun Qu[3], Yang Xiao[4]
[1]*Cyberspace Institute of Advance Technology, Guangzhou University, Guangzhou, China*
[2]*Peng Cheng Laboratory, Shenzhen, China*
[3]*GDCERT/CC, Guangzhou, China*
[4]*Shenzhen Broadton Intellegent Technology Co., Ltd, Shenzhen, China*
{fanjiaxin@e.gzhu.edu.cn, 2112106246@e.gzhu.edu.cn, limohan@gzhu.edu.cn}

***Abstract*—With the widespread deployment of data-driven services, the demand for data volumes continues to grow. At present, many applications lack reliable human supervision in the process of data collection, which makes the collected data contain low-quality data or even malicious data. This low-quality or malicious data make AI systems potentially face much security challenges. One of the main security threats in the training phase of machine learning is data poisoning attacks, which compromise model integrity by contaminating training data to make the resulting model skewed or unusable. This paper reviews the relevant researches on data poisoning attacks in various task environments: first, the classification of attacks is summarized, then the defense methods of data poisoning attacks are sorted out, and finally, the possible research directions in the prospect.**

***Index Terms*—Data Poisoning, Availability Attack, Targeted Attack**

I. Introduction

Currently, machine learning algorithms are widely used in various industries [1]–[3]. At the same time, their security has also received widespread attention from researchers. It has been shown that machine learning is potentially vulnerable to malicious attacks in both the training and inference phases, which may render the model unusable or skewed according to the attacker's intent, with the main threat in the training phase being data poisoning attacks and the main threat in the inference phase being evasion attack. Training phase, but also a few attacks in the inference phase, so some of the backdoor attacks belong to a special kind of data poisoning attack. Data poisoning attacks can be broadly classified into two categories: availability attack and targeted attack, as shown in Fig 1. An availability attack aims to corrupt the model classifier as much as possible; a targeted attack aims to have an impact on specific data points and manipulate specific output results. In addition, two special attacks, clean-label attack [4] and label-flipping attack [5], have been studied according to the control over the training data labels.

Data poisoning attacks focus on how to subliminally change the training data and often occur in the first phase of the entire machine learning lifecycle (i.e., the training phase), the basic principles of data poisoning attacks are shown in Fig 2.

* Mohan Li is corresponding author.

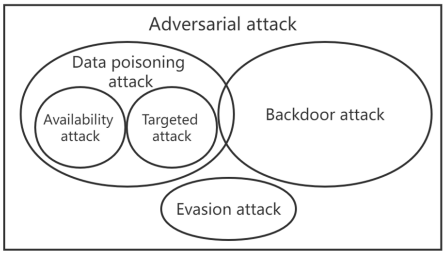

Fig. 1. Three types of adversarial attack.

Where the system first performs data collection, the attacker can then mix the constructed poisoned data into the clean data and then analyze the data features to find vulnerable labels to contaminate (e.g., flipping), thus misleading the model training process afterward. So, data poisoning attacks are a much more realistic and powerful threat.

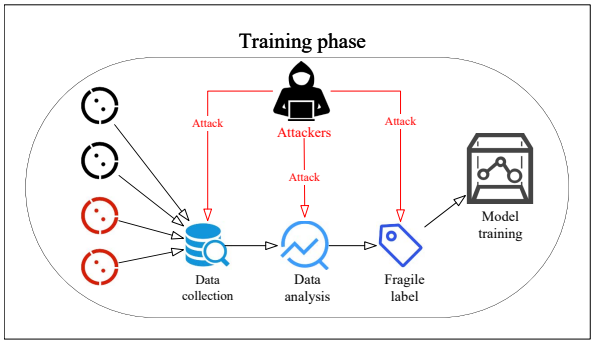

Fig. 2. Basic principles of a data poisoning attack.

The data poisoning literature in recent years has addressed attacks in a variety of environments, in addition to various applications, with threat models ranging from attackers having access to data only to attackers having control over the entire training process. The most common attack environments are recommender systems [6] and crowdsourcing systems [7]. In the study of recommender systems in general, two attack methods are widely used, one is to inject fake item ratings; the other is to inject fake users. In crowdsourcing systems, an attacker can introduce malicious workers and disguise them as normal workers to provide poisoned data, so that the crowdsourcing system gets the wrong aggregated truth value.

978-1-6654-7480-1/22/$31.00 ©2022 IEEE
DOI 10.1109/DSC55868.2022.00014

At the same time, researchers have explored how to defend against data poisoning attacks, such as attack detection and adversarial training for recommender systems, Sybil defense for general crowdsourcing platforms [8], and data aggregation defense for dynamic crowdsourcing systems with online learning algorithms [9]. In addition, various types of attacks [10]–[12], including data poisoning attacks, can also be defended to some extent by analyzing the credibility of data sources or task participants. Most of the existing defense research focuses on how to enhance the robustness of models and the need to design defense methods for specific models. However, some general data quality management techniques can also be used to defend against data poisoning attacks to some extent, and these include data aggregation, data sanitization, and data augmentation. Data aggregation reduces the impact of poisoned data by setting weight parameters to groups of data; data sanitization removes data that deviates significantly from clean data before training the model, and data augmentation add regularity to decision boundaries to prevent misclassification of data.

While attacks are now often evaluated against specific target learning procedures, the fact that the rise of data poisoning attacks has outpaced the development of robust defenses cannot be denied. In this paper, after summarizing the overall research overview of attacks and defenses, we argue that effective defenses should provide reliable protection against a wide range of attacks, that defenders must learn how to detect data poisoning attacks and how to prevent them, that generality and reliability are important in both attacks and defenses, and that we expect to study a wide range of data poisoning attacks and defenses against most scenarios. We expect to investigate a wide range of data poisoning attacks and defenses for most scenarios.

The subsequent sections of the paper are organized as follows: Section II gives some required background knowledge; Section III provides an overview of availability attacks, targeted attacks, and novel subpopulation attacks; Section IV describes how researchers have used existing techniques to defend against poisoning attacks, and Section V summarizes the full paper and discusses related work that follows.

II. PRELIMINARIES

A. Neural Network

With the rise and development of deep learning techniques, numerous systems have begun to introduce deep learning algorithms to train models. Research in deep learning relies more on deep neural networks [13], and recent studies have shown that deep neural networks are particularly vulnerable to data poisoning attacks.

We first consider a training set n where the examples $D = \{x_i, y_i\}_{i=1}^{n}$, each feature vector $x_i \in \chi$ and label $y_i \in \gamma$ are drawn from the data distribution space $\mathcal{D}$. The setting for data poisoning attacks is mostly a multi-classification task, so the label $\gamma = [K]$ for one of the K class problems. The goal of the learning algorithm A in neural networks is to return a model f with parameters θ that correctly classifies as many data points as possible by label and maximizes $\mathbb{E}_{x,y\sim\mathcal{D}} 1(f(x) = y)$, given the entire data set D , a process that is typically done by stochastic gradient descent with differentiable loss function [14]. To approximately minimize the error, the model chooses the SoftMax activation function to finally output the probability of each class K (we write the i class probability as $f(x)_i$). In multi-classification tasks, the categorical cross-entropy loss function ℓ is usually minimized as follows:

$$\ell(X, Y, f) = \frac{1}{|X|} \sum_{i=1}^{|X|} y_i \left(1 - f(x_i)_{y_i}\right) \tag{1}$$

In a neural network, the model f contains a series of linear and non-linear transformations. The linear transformations are called layers and the non-linear transformations are called activation functions. The most widely used activation functions are ReLU, sigmoid, and SoftMax, and the choice of scenario varies depending on the requirements. In standard training, the model parameters θ are initialized randomly. To improve training efficiency, a transfer learning approach can be used where a model trained on a large dataset is used as an initialization for a model on a smaller dataset.

Entire architectures of neural networks have been created to solve a wide variety of tasks and problems. They can be divided into four main categories: standard networks based on perceptrons, convolutional neural networks for a wide range of applications in images, recurrent neural networks for processing time series information and auto-encoders for dimensionality reduction and compression of data.

B. Bilevel Optimization Problem

In data poisoning attacks, adversarial instances (malicious data) are injected into the learner's training dataset to influence the learning algorithm and model according to the attacker's defined target. Earlier work on data poisoning for classical models solves a bilevel optimization problem, but does not precisely address the internal problem. In order to enable bilevel optimization methods to produce optimal attacks against the training set on both simple and complex models, in this section we first discuss the bilevel optimization problem under white-box attacks [15]. Assuming that the classifier is parameterized by θ , the attacker can be represented as a bilevel optimization problem by generating malicious data as follows:

$$\begin{aligned} \arg\max_{\mathcal{D}_p} \quad & \mathcal{W}\left(\mathcal{D}', \theta_p^{\star}\right), \\ s.t. \quad & \theta_p^{\star} \in \operatorname{argmin}_{\theta} \mathcal{L}\left(\mathcal{D}_{tr} \cup \mathcal{D}_p, \theta\right). \end{aligned} \tag{2}$$

where $\mathcal{D}_p$ denotes a set of malicious data, $\mathcal{D}_{tr}$ denotes the training data, and $\mathcal{L}$ is the learning algorithm of the system. External optimization is equivalent to selecting some malicious data to maximize the loss function $\mathcal{W}$ on the initial uncontaminated dataset $\mathcal{D}'$. Internal optimization, on the other hand, retrains the classifier algorithm on a poisoned training set with malicious data $\mathcal{D}_p$. By solving the internal optimization problem, the parameter $\theta_p^{\star}$ implicitly depends on the set of samples for the poisoning attack $\mathcal{D}_p$, in other words, a dataset consisting of malicious and clean data is trained to obtain

$\theta_p^\star$, after which the resulting model classifier attributes the target data to a specific class for the purpose of the attack. Some work in this area has also investigated additional threat models, such as non-targeted attacks, which are highlighted in Section III. In addition, in a black-box setting, attackers do not have access to training data $\mathcal{D}_{tr}$ and estimate the poisoning regression parameters $\theta_p^\star$ by collecting their own alternative training set $\mathcal{D}_{tr}{}'$. Fundamentally, all data poisoning attacks can be described as a bilevel optimization problem, and many effective solutions have been investigated, such as gradient descent [16], gradient alignment or the use of auto-encoders [17] to generate perturbed attack data, etc.

III. DATA POISONING ATTACK

There are various types of adversarial attacks, which can be divided into white-box and black-box attacks in terms of attack environment. An attacker with complete knowledge of the model belongs to a white-box attack; while having no knowledge of the model and only interacting with it through input and output belongs to a black-box attack. In this paper, we do not classify data poisoning attacks in terms of white-box and black-box dimensions.

Data poisoning attacks aim to manipulate the training samples or model architecture to achieve one of the following two objectives: first, to cause misclassification of subsequent input data associated with a specific label; second, to manipulate data predictions for all classes. In addition, there is also recent work proposing a subpopulation attack, which targets between the above two objectives, and an overview of related work is shown in Table I. We divide poisoning attacks into three major categories, and within each major category are subdivided according to technical means and attack targets. In the dimension of the attack target, current approaches can be broadly divided into two categories: one category targets specific machine learning models and its attacks are not limited to the application scenarios of the models; the other category targets specific application scenarios, such as crowdsourcing and knowledge graphs, and the success of its attacks depends on certain features of the application scenarios.

TABLE I
SUMMARY OF DATA POISONING ATTACK TYPES

Types Of Attack	Techniques	Targets	Related Work
Availability Attack	Deep Learning	Crowdsourcing, Crowdsensing	[7,20-21]
		Image classification	[17,22]
	Reinforcement Learning	Graph Neural Network	[23-25]
	Other	Classical machine learning model	[15,18-19]
Targeted Attack	Deep Learning	Image classification	[4,28-30]
		Knowledge graph	[31-32]
	Reinforcement Learning	Autonomous vehicle	[33]
		Recommendation system	[6,34]
	Other	Classical machine learning model	[16,26-27]
Subpopulation Attack	Deep Learning	Image classification, Sentiment analysis	[14]

A. Availability Attack

In an availability attack, attackers seek to maximize the average test loss to cause more damage to the classifier and reduce the overall performance of the model. Earlier, availability attacks mostly targeted classical machine learning models, such as support vector machines (SVMs) [18], Bayesian classifiers [19], and linear regression models [15] for poisoning attack research, but with the rise of big data and deep learning techniques, recent work has focused more on realistic and specific application scenarios, and how attackers can use deep learning and reinforcement learning techniques for most application scenarios to launch data poisoning attacks is the focus of our research.

1) Deep learning: Both crowdsourcing and crowdsensing are new ways of gathering information from the physical world, by posting tasks to participating workers and obtaining the sensory data they provide. It is the open nature of the system that makes it easy for an attacker to reduce the effectiveness of the system by creating or recruiting a group of malicious workers and having them submit malicious data [7]. For the purpose of availability attack and to obtain maximum attack utility, the optimal attack strategy is found by approximating a discrete attack using a sigmoid activation function and then iteratively solving a bilevel optimization problem [20].

In crowdsourcing and crowdsensing systems, the attacker is given the number of malicious workers to create, and the object to observe is assigned to each malicious worker; it is important to find the optimal observation [21]. The final estimated values of the mth object before and after the attack are usually denoted as x_m^{*f} and $\widehat{x}_m^{*f}$ respectively, and the availability attack is represented as an optimization problem as follows:

$$\begin{aligned} \max_{\widetilde{X}} & \sum_{m=1}^{M} \mathbb{H}\left(\hat{x}_m^* \neq x_m^{*f}\right) \\ \text{s.t.} \quad & \left\{\widehat{X}^{*f}, W, \widetilde{W}\right\} = \underset{\hat{X}^{*f}, W, \widetilde{W}}{\arg\min} f\left(\widehat{X}^{*f}, W, \widetilde{W}\right) \\ \text{s.t.} \quad & \sum_{k=1}^{K} \exp\left(-w_k\right) + \sum_{k'=1}^{K'} \exp\left(-\widetilde{w}_{k'}\right) = 1. \end{aligned} \tag{3}$$

where w and $\widetilde{w}$ are the weight parameter for the regular and malicious workers, $\widehat{X}^{*f} = \left\{\widehat{x}_m^{*f}\right\}_{m=1}^{M}$ is the final aggregated value after the attack, and $\mathbb{H}(\cdot)$ is the indicator function. The optimization for the malicious worker data $\widetilde{X}$ is the upper-level problem and the optimization for $\left\{\widehat{X}^{*f}, W, \widetilde{W}\right\}$ given $\widetilde{X}$ is the lower-level problem. x_m^{*f} is the aggregation result obtained by the system before the attack (calculated from the sensory data of the normal workers). Once the normal worker's data is given, it is a constant for the attacker and $\widehat{x}_m^{*f}$ depends on the attack strategy (i.e., $\widetilde{X}$) and may vary depending on the attack strategy.

Image classification is an important application scenario in the field of computer vision. Ji Feng et al. used auto-encoders to add the smallest possible bounded noise to the image dataset [17] to trick machine learning models and change the classification results of image classifiers. Yucheng

Shi et al. proposed forward Curls iteration [22] in a setting where only the model can be queried and each image category score obtained, which improves the diversity of iterative trajectories and the portability of adversarial samples by combining gradient ascent and gradient descent directions and greatly enhances the attack utility of availability attacks by exploiting perturbation robustness through Whey optimization to compress the noise added to the images. However, recent work on image classification has mostly focused on how to change the classification results of target images, which we will focus on in the subsection on the targeted attack.

2) Reinforcement learning: Reinforcement learning provides a powerful approach to solving challenging problems in a variety of domains, but there has been little research on how to poison graph-structured data, and many researchers have explored data poisoning attacks against graph neural networks using reinforcement learning as a technical base. Real-world graph applications, such as advertising and product recommendation, are based on accurate classification of node labels to achieve revenue. In such scenarios, malicious attackers pollute the graph as much as possible to reduce the performance of node classification. Xiao Zang et al. found that the vulnerability of a graph is greatly increased if it contains several malicious nodes [23] that compromise the graph neural network by flipping connections to arbitrary target victims. Previous work on graph attacks has focused on modifying existing graph structures, which is not very feasible in the real world. Instead, it is more practical to inject malicious nodes into existing graphs, which can significantly reduce the performance of classifiers.

Yiwei Sun et al. came up with the idea of using reinforcement learning to perform availability poisoning attacks to manipulate the labels and links of fake nodes [24]. They later conducted an in-depth study, first using Markov Decision Process (MDP) to model the key steps of a node injection attack, such as establishing links between the injected malicious nodes and other nodes, and selecting the labels of the injected nodes. A novel reinforcement learning method for Node Injection Poisoning Attacks (NIPA) [25] is performed by sequentially modifying the labels and links of the injected nodes without changing the existing node connectivity. The implementation introduces a deep Q-network to handle the labels of malicious nodes and their links to other common nodes in the graph and designs an appropriate reward function to guide the agent to reduce the node classification performance of the graph neural network (GNN) during the training process.

B. Targeted Attack

In a targeted attack, the attacker induces a wrong prediction by the classifier by creating false malicious data to distort the true class of certain objects (called target objects) to a specific target answer, which is a stealthier attack. The target answer is usually predetermined by the attacker, and when the targeted attack is performed, if the final classification result of the target object becomes the target answer after the attack, the attack on this object is successful. Otherwise, the attack on this object fails. Early targeted attacks also target classical machine learning models such as logistic regression [16], LASSO [26], and autoregressive models [27]. In this section, in contrast with availability attacks, we focus on targeted attacks using deep learning and reinforcement learning techniques.

1) Deep learning: In the image classification setting, attackers prefer to target specific images rather than indiscriminate attacks, so targeted clean-label attacks are the most common means by which attackers can poison datasets by simply putting targeted malicious images on the network and waiting for data crawling bots, social media platforms, or other unsuspecting victims to access them [28]. Correctly labeled poisoned samples are legally compliant in the human view, but they contain malicious features that trigger targeted misclassification in subsequent inference training.

Shafahi et al. proposed the feature collision (FC) heuristic for clean-label attack [4] that causes the target image to be misclassified by perturbing the training data to collide with the target image in the feature space while viewing the overall model performance fails to detect the presence of the attack. Without controlling the data collection and labeling process, the attacker first selects a target instance from the image dataset t , then selects a base instance from the base class b , and creates a poisoned instance by adding a small adversarial perturbation to b by solving the following optimization problem:

$$x_p = \arg\min_x \|f(x) - f(t)\|_2^2 + \beta \|x - b\|_2^2 \tag{4}$$

$f(x)$ is the feature space representation of the input, the first term of the equation brings the poisoned instances as close as possible to the target instances and embeds them in the distribution of the target class; the second term interferes with the human labeler ($\beta > 0$ parameterizes it) so that the poisoned instances are labeled as base class instances to be added to the training. The fact that the poisoned training data is correctly labeled greatly increases the difficulty of attack detection, and during training tests, the model may incorrectly consider the target instance as being in the base class for the purposes of the targeted attack.

However, FC has limited applicability, and can only result in a single target instance being misclassified, the attacker must also be aware of the feature extraction procedure being used, and the feature extraction procedure cannot be substantially changed after the injection of poisoned data, and the FC attack fails once the victim trains their model from scratch [29]. Later, Jonas Geiping et al. improved the scenario for training from scratch [30], where the victim's model is trained from a random initialization of the poisoned dataset; Hojjat Aghakhani et al. were the first to propose Bullseye Polytope for a multi-target threat model, where poisoned samples are produced in the feature space centered on the target image. The resulting attack is also effective on invisible images of the target while maintaining good baseline test accuracy on non-target images.

Knowledge graph embedding (KGE) is a technique for learning continuous embeddings of entities and relations in the knowledge graph, which has great advantages in accurately predicting missing facts in the knowledge graph and completing them. KGE is very effective in a good environment, Hengtong Zhang et al. proposed a data poisoning attack strategy for knowledge graph embedding data poisoning attack strategy, which effectively manipulates the likelihood of arbitrary target facts in the knowledge graph by adding or removing facts on the knowledge graph [31]. After that Prithu Banerjee et al. consider that every time a poisoning attack is executed, there is a risk of exposure and the value created by this attack is very limited if it is discovered by the victim. Therefore, they proposed the RATA framework [32], which learns low-risk perturbations to maximize the plausibility of manipulating the target facts within the exposure risk.

2) Reinforcement learning: Data poisoning attacks based on reinforcement learning generally focus on overall performance degradation, i.e. availability attacks, while the attacker can slightly modify the dataset before learning to make the learner learn the attacker's chosen target strategy. Harrison Foley et al. introduced a covert target attack for reinforcement learning [33], which selects only a few small perturbations in a specific target state that disrupt the actions of the intelligence while minimally modifying some of the observations, and does not directly control the policy and reward functions. The researchers tried two example attacks, taking the autonomous driving scenario as an example, in the training of car intelligence, the path planning attack is to select a specific road as the attacked state so that the car cannot drive to this road and can pass safely everywhere else; the other attack is for the image training set so that the car keeps running smoothly and safely until it encounters a specific billboard set by the attacker. These subtle modifications make it difficult for defenders to detect if a data poisoning attack has occurred.

In Section I, we mentioned that recommendation systems are vulnerable to data poisoning attacks, the most common attack being the injection of fake users. In news recommendation systems, as each recommendation item (i.e., news), contains more information, this means that a larger space of state actions needs to be perturbed. Xudong Zhang et al. designed a TDP-CP reinforcement learning framework [6] that uses a two-level hierarchical structure and influence estimation to reduce the search space, speed up the computation of rewards, achieve rank manipulation of target news by disturbing the content of some news, and use the limited exposure budget to achieve an attack to increase or decrease the ranking of target news. Recently, Zih-Wun Wu et al. proposed two reward mechanisms for knowledge graph recommender systems to determine the best combination of perturbations [34] that can recommend a specific pre-selected item to a larger number of people after an attack.

C. Subpopulation Attack

Subpopulation attack is a middle ground between availability and targeted data poisoning attack proposed by Matthew Jagielski et al [14]. In an availability attack, the attacker needs to control a larger portion of the training data to influence the model, whereas a targeted attack requires the attacker to identify the target point in advance and has significant limitations. A subpopulation attack is suitable for large and diverse datasets, where the attacker aims to impair the performance of the classifier on a specific subpopulation while keeping the performance unchanged for the remaining subpopulations. The advantage of this novel poisoning attack is that it does not require advance access to the victim's architecture, algorithm parameters, and actual training data, but only an auxiliary dataset, as shown in Fig 3. The vulnerable subpopulations are first identified by two subpopulation selection methods, CLUSTER MATCH or FEATURE MATCH, and then the poisoning attack is generated by label flipping or attack optimization. Finally, the poisoned data and clean data are mixed to influence the learning algorithm and model training process afterward.

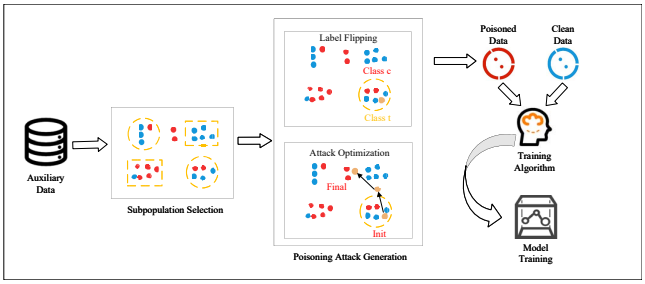

Fig. 3. Subpopulation attack framework.

A subpopulation attack can be considered as an availability attack or targeted attack in special cases, such as when a filter function is defined for the entire auxiliary data set corresponding to the former; filter selection of a single point (or a small group of points) corresponding to the latter. Compared to an availability attack, a subpopulation attack is stealthy and difficult to detect; compared to a targeted attack, a subpopulation attack has a greater impact and can produce greater damage.

IV. DATA POISONING DEFENSE

The growing number of data poisoning attacks is one of the most worrying, with many defenses against data poisoning either failing in the face of increasingly powerful attacks or significantly degrading performance. Whether using deep learning or reinforcement learning techniques, poisoning attacks in the environment of systems that are closely related to humans have a high potential to threaten human lives and property. In the following, we summarize the development of defense techniques, as shown in Table II. We divide the current defenses into four categories, and in each category, we summarize the commonly used technical tools and their application stages against data poisoning attacks.

TABLE II
SUMMARY OF DATA POISONING DEFENSE METHODS

Types Of Defense	Techniques	Applications	Related Work
Data aggregation	Truth discovery	Data analysis	[8,35]
	MWA, MIE		[9,36-37]
	DPA		[38]
Data sanitization	DPIF, L2 and $Slab$	Data collection	[5,39-43]
Data augmentation	mixup and CutMix	Data analysis	[44-45]
Robustness enhancement	Influence function	Model training	[43,46]
	SGD		[40,47]
	Decision tree		[48-49]

A. Data Aggregation

In the original crowdsourcing and crowdsensing systems, sensory data from workers was aggregated using majority voting [8]; however, not all workers involved were reliable and malicious workers may have been present. In order to identify true values from the dataset, truth discovery algorithms have been extensively investigated. Traditional truth discovery methods usually assign a trust weight to each worker to estimate the reliability of the data source and predict the true value based on the reliability, and so are robust to data poisoning attacks carried out by malicious workers. The Conflict Resolution on Heterogeneous (CRH) and Gaussian Truth Model (GTM) are two of the most commonly used and advanced truth discovery algorithms applied to typed and numerical data respectively. In addition, the Dawid-Skene model can also tolerate data poisoning attacks to some extent by using the Expectation-Maximization (EM) algorithm for Maximum Likelihood Estimation (MLE) [35], jointly estimating the reliability of each worker and performing weighted aggregation.

However, with the development of availability attacks and targeted attacks, attackers successfully disguise malicious workers by crafting malicious data, which greatly reduces the performance of existing truth discovery algorithms [36]. Minghong Fang et al. studied poisoning attacks and proposed two defense mechanisms, Median-of-Weighted-Average (MWA) and Maximize Influence of Estimation (MIE), to mitigate the impact of poisoning attacks [37]. MIE assumes that the server knows that the system is under attack and knows the number of malicious workers, so it has better defense performance than MWA, but neither can cope with a high percentage of malicious workers. Later, Yuxi Zhao et al. integrated median and MIE-based data aggregation for the defense of a dynamic crowdsourcing system [9].

Deep Partition Aggregation (DPA) is an aggregation-based authentication defense method that directly partitions the training set into disjoint subsets. Recently, Wenxiao Wang et al. improved on and extended this method [38] by first partitioning the training set into smaller disjoint subsets and then combining their copies to build larger (but not disjoint) subsets for the classifier, reducing the worst effects of poisoned data. The limited data aggregation approach defends to a large extent against generic data poisoning attacks.

B. Data Sanitization

Data sanitization is a common defense method for data poisoning attacks, removing data that deviates significantly from clean data before training the model. Mohan Li et al. proposed a two-steps DPIF framework [39], which first detects false candidates using data quality rules, then uses clustering to find potential false users, and cleanses the malicious data generated by false users by analyzing data quality.

Outlier detection is often used to remove noise from data [40], but the method is not effective in removing all counteracting noise in the face of an attacker's carefully designed malicious data points. Steinhardt et al. propose defenses with different outlier detection rules [41], which enhance the defense to some extent. Other existing defenses for data sanitization are L2, $Slab$ and $loss$. The L2 defense discards points away from the data center directly [5], while the $Slab$ defense first projects all data points onto a line between two classes of data centers and then discards points away from the corresponding data center. Similarly, the $loss$ defense discards data points that do not match the model on the entire dataset.

However, in recent work, Pang Wei Koh et al. developed three attacks that can bypass common data sanitization defenses, including anomaly detectors based on nearest neighbors, training loss and singular-value decomposition [42], using a small amount of poisoned data to achieve attacks that degrade the performance of classified datasets, highlighting the need to develop more robust defenses. Sanjay Seetharaman et al. considered combining the $Slab$ defense with an influence function [43] to obtain an effective defense strategy during their study of data poisoning defenses against online learning algorithms, but found that the minimization of the impact objective function sometimes affected clean data points, resulting in information loss, and that this strategy was not yet complete.

C. Data Augmentation

Research has shown that more sophisticated data augmentation methods can improve the performance of models by adding regularity to decision boundaries and preventing target objects from being incorrectly classified into specified classes, and to some extent successfully defending against data poisoning attacks [44]. Eitan Borgnia et al. proposed more robust data augmentation methods mixup and CutMix that can significantly resist backdoor attacks and poisoning attacks without sacrificing model performance [45].

Mixup combines the training data obtained from random sampling in pairs of convex combinations and uses the corresponding labeled convex combinations, forcing the corresponding assignments to prevent the presence of memory for corrupted labels. The method regularizes class boundaries and removes small non-convex regions (poisoned data surrounded by clean data), improving generalization ability. CutMix generates training data by combining random blocks of an image overlaid onto other images, after which the

labels are mixed with the image blocks in proportion to their area to correctly classify the images from a local view. The researchers designed adaptive attack scenarios, where the attacker is assumed to know the defender and can optimize the attack strategy for that defense, and the experimental results show that both data augmentation methods achieve good classification accuracy even under poisoning attacks, giving the system a more powerful localization capability.

Modern data augmentation methods mitigate the threat of data poisoning attacks that begin with the training of the model, even if the attacker is always optimizing his attack strategy. We consider the design of appropriate data augmentation methods for specific poisoned data or attack methods to be a fruitful direction for future research.

D. Robustness Enhancement

Influence functions are a classic technique in robust statistics [46] and have a variety of uses in linear models and neural networks: understanding model behavior, debugging models, detecting errors in datasets, etc. The core of the influence function is to measure the effect of local changes, and the combination of data sanitization techniques and influence functions, mentioned in Section B, can minimize the utility of attacks on poisoned data, enhancing model robustness to some extent [43]. We believe that applying the impact function to defend against poisoning attacks can improve the reliability and fairness of the system environment.

Studies have shown that neural networks trained using stochastic gradient descent (SGD) algorithms perform well in various aspects, such as optimization and generalization prediction. However, the neural network can be very vulnerable and highly flawed if attacked by data poisoning attacks. Yunjuan Wang et al. investigated the robustness against data poisoning attacks and found that in a two-layer neural network with a ReLU activation function, the SGD algorithm can withstand a certain level of poisoning attacks, including special clean-label attacks and label-flipping attacks, demonstrating that SGD is robustness in learning training data with added adversarial perturbations [47]. However, the limitation is that the width of the neural network is sufficient but not too large, and the generalization error shows a U-shaped curve as a function of network width; they will continue to explore the robustness of SGD under ultra-wide neural networks in their next work. In addition, the differential privacy-based DP-SGD is also successful in defending against poisoning attacks by limiting the gradient size and minimizing the difference in direction by random noise addition [40].

Decision tree algorithms are often used to extract models and knowledge to find out how certain variables are associated with important data classes and represent them as classification rules. Shihao Chang et al. proposed a cloud-based trust management scheme (CbTMS) to detect Sybil attacks in MCS networks [48], using a decision tree algorithm to verify the nodes covered in the MCS network, thus effectively detecting malicious Sybil nodes in the network. Later, Samuel Drews et al. proposed a reliable verification technique based on abstract interpretation [49]. The decision tree algorithm was trained with the antidote tool, trained on a large space of poisoned datasets, and the results showed that for a given input, the corresponding predictions do not change regardless of whether the training set is tampered with, demonstrating the robustness of the antidote against data poisoning in the decision tree algorithm.

V. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we provide a comprehensive discussion of the causes, hazards, types, and countermeasures of data poisoning as an adversarial attack. Firstly, it is noted that because machine learning models are scaling in capability and size, their vulnerability is exposed to attackers, so there is a rapid proliferation of data poisoning attacks and defenses; secondly, the problem of neural networks and bilevel optimization is explained in detail. The core of this paper is a summary of the availability, targeted and novel subpopulation of data poisoning attacks and four means of defending against poisoning attacks. In addition, we find that there are many more technologies that suffer from data poisoning attacks, as well as defenses that we have not mentioned.

In future work, we will continue to investigate flaws and improvements in poisoning attacks and defense methods.

- On the attack side, current algorithms for data poisoning attacks based on bilevel optimization are computationally expensive and hardly support large-scale engineering attack tasks; many attackers assume that they fully understand the entire dataset, which can be very limiting in practical scenarios, and most work on attacks focuses on forcing misclassification of specific target objects or all data at the same time, and existing poisoning attack methods do not seem to be reliable when going to attack network architectures that differ from the original experimental setup.
- On the defense side, the trade-off between accuracy, security, and data privacy is an issue, and according to our current approach, it is not possible to keep the model system safe from poisoning attacks while maintaining accuracy and protecting data privacy. Furthermore, it is undeniable that most defenses nowadays are unable to cope with increasingly powerful poisoning attacks, and the defense methods regarding specific data poisoning attacks are not perfect.

To address the shortcomings of existing attacks and defense efforts, we want to look for attacks that can be transferred to a wide range of training hyperparameters, maximizing the utility of the attack while ensuring that it is undetected by defenders; we want to combine deep learning and multi-agent reinforcement learning to design more sophisticated attacks that enable a crowdsourcing environment where different malicious workers can create different malicious data that over time change, a situation in which the attacker may do more damage. Subsequently, from the attacker's perspective, we attempt to create defenses against most forms of poisoning attacks, hoping to design efficient and practical defenses

against practical applications (e.g., federated learning) with fewer data and computational requirements. Furthermore, we believe that in reinforcement learning techniques, if both attacking and defending agents can detect each other, then the whole environment is dynamic, and it is a worthwhile direction to investigate how the attacker and defender make decisions and execute at that point.

ACKNOWLEDGMENT

This work is funded by the Guangdong Basic and Applied Basic Research Foundation (No.2021A1515012307, 2020A1515010450), Guangzhou Basic and Applied Basic Research Foundation (No. 202102021207, 202102020867), the National Key Research and Development Plan (Grant No. 2020YFB2009503), the National Natural Science Foundation of China (No. 62072130), Guangdong Province Universities and Colleges Pearl River Scholar Funded Scheme (2019), and Guangdong Higher Education Innovation Group (No. 2020KCXTD007) and Guangzhou Higher Education Innovation Group (No. 202032854), Consulting project of Chinese Academy of Engineering(2021-HYZD-8-3), the Major Key Project of PCL(Grant No. PCL2021A02, PCL2022A03, PCL2021A09).

REFERENCES

[1] Jordan M I, Mitchell T M. Machine learning: Trends, perspectives, and prospects[J]. Science, 2015, 349(6245): 255-260.
[2] Butler K T, Davies D W, Cartwright H, et al. Machine learning for molecular and materials science[J]. Nature, 2018, 559(7715): 547-555.
[3] Liakos K G, Busato P, Moshou D, et al. Machine learning in agriculture: A review[J]. Sensors, 2018, 18(8): 2674.
[4] Shafahi A, Huang W R, Najibi M, et al. Poison frogs! targeted clean-label poisoning attacks on neural networks[J]. Advances in neural information processing systems, 2018, 31.
[5] Chan P P K, He Z, Hu X, et al. Causative label flip attack detection with data complexity measures[J]. International Journal of Machine Learning and Cybernetics, 2021, 12(1): 103-116.
[6] Zhang X, Wang Z, Zhao J, et al. Targeted Data Poisoning Attack on News Recommendation System[J]. arXiv, 2022.
[7] Wang G, Wang T, Zheng H, et al. Man vs. machine: Practical adversarial detection of malicious crowdsourcing workers[C]//23rd USENIX Security Symposium (USENIX Security 14). 2014: 239-254.
[8] Yuan D, Li G, Li Q, et al. Sybil defense in crowdsourcing platforms[C]//CIKM, 2017: 1529-1538.
[9] Zhao Y, Gong X, Lin F, et al. Data Poisoning Attacks and Defenses in Dynamic Crowdsourcing with Online Data Quality Learning[J]. IEEE Transactions on Mobile Computing, 2021.
[10] Z. Tian, X. Gao, S. Su and J. Qiu. Vcash: A Novel Reputation Framework for Identifying Denial of Traffic Service in Internet of Connected Vehicles[J]. IEEE Internet of Things Journal, 2019.
[11] Z. Tian, M. Li, M. Qiu, Y. Sun, S. Su. Block-DEF: A Secure Digital Evidence Framework using Blockchain[J]. Information Sciences, 2019.
[12] Y. Sun, Z. Tian, M. Li, S. Su, X. Du, and M. Guizani. Honeypot Identification in Softwarized Industrial Cyber-Physical Systems[J]. IEEE Transactions on Industrial Informatics, 2020.
[13] Abiodun O I, Jantan A, Omolara A E, et al. State-of-the-art in artificial neural network applications: A survey[J]. Heliyon, 2018, 4(11): e00938.
[14] Jagielski M, Severi G, Pousette Harger N, et al. Subpopulation data poisoning attacks[C]//CCS, 2021: 3104-3122.
[15] Jagielski M, Oprea A, Biggio B, et al. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning[C]//2018 IEEE Symposium on Security and Privacy, 2018: 19-35.
[16] Mei S, Zhu X. Using machine teaching to identify optimal training-set attacks on machine learners[C]//AAAI, 2015.
[17] Feng J, Cai Q Z, Zhou Z H. Learning to confuse: generating training time adversarial data with auto-encoder[J]. NIPS, 2019, 32.
[18] Biggio B, Nelson B, Laskov P. Poisoning attacks against support vector machines[J]. arXiv preprint arXiv:1206.6389, 2012.
[19] Nelson B, Barreno M, Chi F J, et al. Exploiting machine learning to subvert your spam filter[J]. LEET, 2008, 8(1-9): 16-17.
[20] Miao C, Li Q, Xiao H, et al. Towards data poisoning attacks in crowd sensing systems[C]//MobiHoc, 2018: 111-120.
[21] Li M, Sun Y, Lu H, et al. Deep reinforcement learning for partially observable data poisoning attack in crowdsensing systems[J]. IEEE Internet of Things Journal, 2019, 7(7): 6266-6278.
[22] Shi Y, Wang S, Han Y. Curls & whey: Boosting black-box adversarial attacks[C]//CVPR, 2019: 6519-6527.
[23] Zang X, Xie Y, Chen J, et al. Graph universal adversarial attacks: A few bad actors ruin graph learning models[J]. arXiv, 2020.
[24] Sun Y, Wang S, Tang X, et al. Node injection attacks on graphs via reinforcement learning[J]. arXiv preprint arXiv:1909.06543, 2019.
[25] Sun Y, Wang S, Tang X, et al. Adversarial attacks on graph neural networks via node injections: A hierarchical reinforcement learning approach[C]//WWW, 2020: 673-683.
[26] Xiao H, Biggio B, Brown G, et al. Is feature selection secure against training data poisoning?[C]//PMLR, 2015: 1689-1698.
[27] Alfeld S, Zhu X, Barford P. Data poisoning attacks against autoregressive models[C]//AAAI, 2016.
[28] Zhu C, Huang W R, Li H, et al. Transferable clean-label poisoning attacks on deep neural nets[C]//PMLR, 2019: 7614-7623.
[29] Huang W R, Geiping J, Fowl L, et al. Metapoison: Practical general-purpose clean-label data poisoning[J]. NIPS, 2020, 33: 12080-12091.
[30] Geiping J, Fowl L, Huang W R, et al. Witches' brew: Industrial scale data poisoning via gradient matching[J]. arXiv, 2020.
[31] Zhang H, Zheng T, Gao J, et al. Data poisoning attack against knowledge graph embedding[J]. arXiv preprint arXiv:1904.12052, 2019.
[32] Banerjee P, Chu L, Zhang Y, et al. Stealthy targeted data poisoning attack on knowledge graphs[C]//2021 IEEE 37th International Conference on Data Engineering, 2021: 2069-2074.
[33] Foley H, Fowl L, Goldstein T, et al. Execute Order 66: Targeted Data Poisoning for Reinforcement Learning[J]. arXiv, 2022.
[34] Wu Z W, Chen C T, Huang S H. Poisoning attacks against knowledge graph-based recommendation systems using deep reinforcement learning[J]. Neural Computing and Applications, 2022, 34(4): 3097-3115.
[35] Miao C, Li Q, Su L, et al. Attack under disguise: An intelligent data poisoning attack mechanism in crowdsourcing[C]//WWW, 2018: 13-22.
[36] Huang Z, Pan M, Gong Y. Robust truth discovery against data poisoning in mobile crowdsensing[C]//GLOBECOM, 2019: 1-6.
[37] Fang M, Sun M, Li Q, et al. Data poisoning attacks and defenses to crowdsourcing systems[C]//Proceedings of the Web Conference, 2021: 969-980.
[38] Wang W, Levine A, Feizi S. Improved Certified Defenses against Data Poisoning with (Deterministic) Finite Aggregation[J]. arXiv, 2022.
[39] Li M, Sun Y, Su S, et al. DPIF: a framework for distinguishing unintentional quality problems from potential shilling attacks[J]. Computers, Materials and Continua, 2019.
[40] Goldblum M, Tsipras D, Xie C, et al. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses[J]. TPAMI, 2022.
[41] Steinhardt J, Koh P W W, Liang P S. Certified defenses for data poisoning attacks[J]. NIPS, 2017, 30.
[42] Koh P W, Steinhardt J, Liang P. Stronger data poisoning attacks break data sanitization defenses[J]. Machine Learning, 2022, 111(1): 1-47.
[43] Seetharaman S, Malaviya S, Vasu R, et al. Influence based defense against data poisoning attacks in online learning[C]//COMSNETS, 2022.
[44] Schwarzschild A, Goldblum M, Gupta A, et al. Just how toxic is data poisoning? a unified benchmark for backdoor and data poisoning attacks[C]//PMLR, 2021: 9389-9398.
[45] Borgnia E, Cherepanova V, Fowl L, et al. Strong data augmentation sanitizes poisoning and backdoor attacks without an accuracy trade-off[C]//ICASSP, 2021: 3855-3859.
[46] Koh P W, Liang P. Understanding black-box predictions via influence functions[C]//PMLR, 2017: 1885-1894.
[47] Wang Y, Mianjy P, Arora R. Robust Learning for Data Poisoning Attacks[C]//PMLR, 2021: 10859-10869.
[48] Chang S H, Chen Z R. Protecting mobile crowd sensing against sybil attacks using cloud based trust management system[J]. Mobile Information Systems, 2016, 2016.
[49] Drews S, Albarghouthi A, D'Antoni L. Proving data-poisoning robustness in decision trees[C]//PLDI, 2020: 1083-1097.