Backdoor Learning: A Survey

Yiming Li©, Graduate Student Member, IEEE, Yong Jiang©, Zhifeng Li©, Senior Member, IEEE, and Shu-Tao Xia©

Abstract—Backdoor attack intends to embed hidden backdoors into deep neural networks (DNNs), so that the attacked models perform well on benign samples, whereas their predictions will be maliciously changed if the hidden backdoor is activated by attacker-specified triggers. This threat could happen when the training process is not fully controlled, such as training on third-party datasets or adopting third-party models, which poses a new and realistic threat. Although backdoor learning is an emerging and rapidly growing research area, there is still no comprehensive and timely review of it. In this article, we present the first comprehensive survey of this realm. We summarize and categorize existing backdoor attacks and defenses based on their characteristics, and provide a unified framework for analyzing poisoning-based backdoor attacks. Besides, we also analyze the relation between backdoor attacks and relevant fields (i.e., adversarial attacks and data poisoning), and summarize widely adopted benchmark datasets. Finally, we briefly outline certain future research directions relying upon reviewed works. A curated list of backdoor-related resources is also available at https://github.com/THUYimingLi/backdoor-learning-resources.

Index Terms—AI security, backdoor attack, backdoor defense, backdoor learning, deep learning.

I. INTRODUCTION

O VER the past decade, deep neural networks (DNNs) have been successfully applied in many mission-critical tasks, such as face recognition, autonomous driving, and so on. Accordingly, its security is of great significance and has attracted extensive concerns. One well-studied example is adversarial examples [1]–[6], which explored the adversarial vulnerability of DNNs at the inference stage. Compared with the inference stage, the training of DNNs contains more steps, including data collection, data cleaning and preprocessing, feature engineering, model selection and construction, training, model evaluation and fine-tuning, model saving, and model deployment. More steps mean more chances for the attackers. Meanwhile, it is well known that the powerful capability of DNNs significantly depends on a large amount of training data and computing resources. To reduce the training costs, users may choose to adopt third-party datasets, rather than collect the training data by themselves, since there are many publicly available datasets; users may also train DNNs based on third-party platforms (e.g., cloud computing platforms), rather than train DNNs locally; users may even directly use third-party backbones or pretrained models. The cost of convenience is the loss of control during the training stage, which may further enlarge the security risk of training DNNs. One typical threat to the training stage is the backdoor attacks,1 which is the main focus of this survey.

In general, backdoor attackers intend to embed hidden backdoors in DNNs during the training process, so that the attacked DNNs behave normally on benign samples whereas their predictions will be maliciously and consistently changed if hidden backdoors are activated by attacker-specified trigger patterns. These attacks may lead to serious consequences in mission-critical applications. For example, the adversaries could cause a backdoored automated driving system to incorrectly identify traffic signs attached with the backdoor trigger, causing traffic accidents. Currently, poisoning training samples [7]–[9] are the most straightforward and widely adopted method to encode backdoor functionality during the training process. For example, as demonstrated in Fig. 1, some training samples could be consistent with the target label [10]–[12] and the ground-truth label of poisoned samples could also be consistent with the target label [13]–[15], which increases the stealthiness of backdoor attacks. Except by directly poisoning the training samples, the hidden backdoor could also be embedded through transfer learning [16]–[18], directly modifying model parameters [19]–[21], and adding extra malicious modules [22]–[24]. In other words, backdoor attacks may happen at all steps involved in the training process.

To alleviate the backdoor threat, different defenses were proposed. In general, those methods can be divided into two main categories, including empirical backdoor defenses and certified

1. Backdoor is also commonly called neural trojan or trojan. In this survey, we use “backdoor” instead of other terms since it is most frequently used.
backdoor defenses. Empirical backdoor defenses [25]–[27] are proposed based on some observations or understandings of existing attacks and have decent performance in practice; however, their effectiveness has no theoretical guarantee and may probably be bypassed by some adaptive attacks. In contrast, the validity of certified backdoor defenses [28]–[30] is theoretically guaranteed under certain assumptions, whereas its performance is generally weaker than that of empirical defenses in practice since those assumptions are usually unsatisfied. How to better defend against backdoor attacks is still an important open question.

Given the fast development of backdoor attacks and defenses, in this survey, we intend to provide a timely overview and discussion of existing methods. Different from concurrent papers which summarized only limited research [31]–[33] or classified existing methods simply by the adversary capabilities [34]–[36], we provide a brief yet comprehensive review as well as the taxonomy for existing methods based on their characteristics and properties. To the best of our knowledge, this is the first systematic taxonomy for backdoor attacks and defenses. With this taxonomy, researchers and practitioners can better identify the properties and limitations of each method to facilitate the design of more advanced methods. We hope that our survey can inspire more understanding of backdoor defenses.

The illustration of the main technical terms is shown in Fig. 2.

II. PRELIMINARIES

A. Definition of Technical Terms

In this section, we briefly describe and explain common technical terms used in backdoor learning. We will follow the same definition of terms in this remaining article.

1) **Benign model** refers to the model trained under benign settings.

2) **Infected model** refers to the model with hidden backdoor(s).

3) **Poisoned sample** is the modified training sample used in poisoning-based backdoor attacks for embedding backdoor(s) in the model during the training process.

4) **Trigger** is the pattern used for generating poisoned samples and activating the hidden backdoor(s).

5) **Attacked sample** indicates the malicious testing sample containing backdoor trigger(s).

6) **Attack scenario** refers to the scenario that the backdoor attack might happen. Usually, it happens when the training process is inaccessible or out of control by the user, such as training with third-party datasets, training through third-party platforms, or adopting third-party models.

7) **Source label** indicates the ground-truth label of a poisoned or an attacked sample.

8) **Target label** is the attacker-specified label. The attacker intends to make all attacked samples be predicted as the target label by the infected model.

9) **Attack success rate (ASR)** denotes the proportion of attacked samples, which are successfully predicted as the target label by the infected model.

10) **Benign accuracy (BA)** indicates the accuracy of benign test samples predicted by the infected model.

11) **Attacker’s goal** describe what the backdoor attacker intends to do. In general, the attacker intends to design an infected model that performs well on the benign testing sample while achieving high ASR.

12) **Capacity** defines what the attacker/defender can and cannot do to achieve their goal.

13) **Attack/defense approach** illustrates the process of the designed backdoor attack/defense.

The illustration of the main technical terms is shown in Fig. 2.

B. Classical Scenarios and Corresponding Capacities

In this section, we introduce three classical real-world scenarios that which backdoor threats could occur, and their corresponding attacker’s and defender’s capacities. More details are summarized in Table I and illustrated as follows.

**Scenario 1 (Adopt Third-Party Datasets):** In this scenario, attackers provide the poisoned dataset to users directly or through the Internet. Users will adopt the (poisoned) dataset to alleviate the backdoor threat.

**Scenario 2 (Training with Attackers):** In this scenario, benign models are trained with poisoned data, and users train their models using the poisoned dataset. The attacker can only manipulate the dataset, whereas cannot modify the model, the training schedule, and the inference pipeline. In contrast, defenders can manipulate everything in this scenario. For example, they can clean up the (poisoned) dataset to alleviate the backdoor threat.

**Scenario 3 (poisoning-based backdoor attacks):** In this example, the trigger is a black square on the bottom right corner and the target label is “0.” Part of the benign training images is modified to have the trigger stamped, and their label is re-assigned as the attacker-specified target label. Accordingly, the trained DNN is infected, which will recognize attacked images (i.e., test images containing backdoor trigger) as the target label while still correctly predicting the label for the benign test images.
Fig. 2. Illustration of technical terms. (a) Attack configurations. (b) Training process. (c) Inference process.

Fig. 3. Taxonomy of poisoning-based backdoor attacks with different categorization criteria. In this figure, the red boxes represent categorization criteria, while the blue boxes indicate attack subcategories. Please refer to Table II for more technical details.

TABLE I
THREE CLASSICAL SCENARIOS AND CORRESPONDINGLY ATTACKER’S AND DEFENDER’S CAPACITIES. FROM TOP TO BOTTOM, THE ATTACKER’S CAPACITIES GRADUALLY INCREASE, WHILE THE DEFENDER’S ONES GRADUALLY DECREASE

| Scenario 1, Capacity | Attackers | Defenders |
|----------------------|-----------|-----------|
| Adopt Third-Party Datasets | ● ○ ○ ○ | ● ● ● ● |
| Adopt Third-Party Platforms | ● ● ○ ○ | ○ ○ ○ ○ |
| Adopt Third-Party Models | ● ● ○ ○ | ○ ○ ○ ○ |

1: controllable; ○: uncontrollable; ●: partly controllable (It is partly uncontrollable for defenders when using the third-party model’s API, while it is controllable when adopting pre-trained models).

Scenario 2 (Adopt Third-Party Platforms): In this scenario, users provide their (benign) dataset, model structure, and training schedule to an untrusted third-party platform (e.g., Google Cloud) to train their model. Although the benign dataset and training schedule are provided, the attacker (i.e., the malicious platform) can modify them during the actual training process. However, the attacker cannot change the model structure otherwise users will notice the attack. On contrary, defenders cannot control the training set and schedule, while they can modify the trained model to alleviate the attack. For example, they can fine-tune it on a small local benign dataset.

Scenario 3 (Adopt Third-Party Models): In this scenario, attackers provide trained infected DNNs through the application programming interface (API) or the Internet. Attackers can change everything except for the inference pipeline. For example, the user can introduce a preprocessing module on the test image before the prediction, which is out of control by the attackers. For the defenders, they can control the inference pipeline and also the model when its source files are provided; however, if they can only get access to the model API, they cannot modify the model.

In particular, attackers’ capacities increase while defenders’ capacities decrease from Scenario 1 to Scenario 3. In other words, attacks designed for a previous scenario could also occur in the following ones; similarly, defenses designed for a later scenario could also be used in previous ones.

III. POISONING-BASED BACKDOOR ATTACKS

In the past four years, many backdoor attacks were proposed. In this section, we first propose a unified framework to analyze existing poisoning-based attacks toward image classification, based on the understanding of attack properties. After that, we summarize and categorize existing poisoning-based attacks in detail, based on the proposed framework. Attacks for other tasks or paradigms and the positive applications of backdoor attacks are also discussed at the end. The example of poisoned samples generated by different types of backdoor attacks is shown in Fig. 4.

A. Unified Framework of Poisoning-Based Attacks

Poisoning-based backdoor attacks can be categorized based on different property-related criteria, as shown in Fig. 3 and summarized in Table II. More details are as follows.

We denote the classifier as \( f_w : \mathcal{X} \rightarrow [0, 1]^K \), where \( w \) is the model parameters, \( \mathcal{X} \subseteq \mathbb{R}^d \) being the instance space, and \( \mathcal{Y} = \{1, 2, \ldots, K\} \) being the label space. \( f(x) \) indicates the posterior vector with respect to \( K \) classes, and \( C(x) = \arg \max f_w(x) \) denotes the predicted label. Let \( G_t : \mathcal{X} \rightarrow \mathcal{X} \)
indicates the attacker-specified poisoned image generator with trigger pattern $t$, and $S : \mathcal{Y} \to \mathcal{Y}$ is the attacker-specified label shifting function. Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ indicates a benign dataset, three classical risks (with respect to $\mathcal{D}$) involved in existing backdoor attacks can be defined as follows.

Definition 1 (Standard, Backdoor, and Perceivable Risk):
1) The standard risk $R_s$ measures whether the infected model $C$ can correctly predict benign samples. That is,

$$R_s(\mathcal{D}) = \mathbb{E}_{(x,y) \sim \mathcal{D}} \mathbb{I}(C(x) \neq y)$$

where $\mathcal{D}$ is the distribution behind $\mathcal{D}$ and $\mathbb{I}(-)$ is the indicator function. $\mathbb{I}(\{A\}) = 1$ if and only if the event “$A$” is true.

2) The backdoor risk $R_b$ indicates whether backdoor attackers can successfully achieve their malicious purposes in predicting attacked samples. That is,

$$R_b(\mathcal{D}) = \mathbb{E}_{(x,y) \sim \mathcal{D}} \mathbb{I}(C(x') \neq S(y))$$

where $x' = G_t(x)$ is the attacked image.

3) The perceivable risk $R_p$ denotes whether the poisoned sample is detectable (by human or machine), That is,

$$R_p(\mathcal{D}) = \mathbb{E}_{(x,y) \sim \mathcal{D}} D(x')$$

where $D(\cdot)$ is an indicator function. $D(x') = 1$ if and only if $x'$ can be detected as the malicious sample.

Given a benign training set $\mathcal{D}_b$, existing poisoning-based backdoor attacks can be summarized in a unified framework based on aforementioned definitions, as follows:

$$\min_{t,w} \mathbb{E}_{(x,y) \sim \mathcal{D}_b} \left\{ (\mathbb{I}(C(x) \neq y) + \mathbb{I}(C(x') \neq S(y)) + \lambda_1 \cdot \mathbb{I}(C(x') \neq S(y)) + \lambda_2 \cdot D(x') \right\}, \quad \text{where } t \in T \text{ and } x' = G_t(x)$$

In particular, this framework can be easily generalized toward other tasks, such as speech recognition, as well. Since there were many different types of tasks and their papers were limited, this generalization is out of the scope of this survey.

B. Evaluation Metrics

To evaluate the performance of backdoor attacks in the image classification, two classical metrics are usually adopted, including: 1) BA and 2) ASR, as defined in Section II-A. The higher the BA and ASR, the better the attack. Besides, the smaller the poisoning rate and the perturbation between the benign image and the poisoned image, the more stealthy the attack.

C. Attacks for Image and Video Classification

1) BadNets: Gu et al. [7] introduced the first backdoor attack in deep learning by poisoning some training samples. This method was called BadNets. Specifically, as demonstrated in Fig. 1, its training process consists of two main parts, including: 1) generate some poisoned images via stamping the backdoor trigger onto selected benign images to achieve poisoned sample $(x', y_i)$, associated with the attacker-specified target label $y_i$ and 2) release the poisoned training set containing both poisoned and benign samples to victims for training their own models. Accordingly, the trained DNN will be infected, which performs well on benign testing samples, similar to the model trained using only benign samples; however, if the same trigger is contained in an attacked image, then its prediction will be changed to the target label. This attack could happen in all scenarios described in Section II-B, and therefore, is a serious security threat. BadNets is the representative of visible attacks, which opened the era of this field. Almost all follow-up poisoning-based attacks were carried out based on this method.

2) Invisible Backdoor Attacks: Chen et al. [10] first discussed the invisibility requirement of poisoning-based backdoor attacks. They suggested that the poisoned image should be indistinguishable compared from its benign version to evade human inspection. To fulfill this requirement, they proposed a blended strategy, which generated poisoned images by blending the backdoor trigger with benign images instead of by stamping (as adopted in BadNets [7]). Besides, they showed that even adopting a random noise with a small magnitude as the backdoor trigger can still create the backdoor successfully, which further reduces the risk of being detected.
After that, there was a series of works dedicated to the research of invisible backdoor attacks. Turner et al. [13] proposed to perturb the pixel values of benign images by a backdoor trigger amplitude instead of by replacing the corresponding pixels with the chosen pattern. Zhong et al. [37] adopted the universal adversarial attack [38] to generate backdoor triggers, which minimizes the $\ell^2$ norm of the perturbation to ensure invisibility. After that, [11], [39], and [40] proposed to regularize the $\ell^p$ norm of the perturbation when optimizing the backdoor trigger. Liu et al. [8] proposed to adopt a common phenomenon (i.e., the reflection) as the trigger for stealthiness. Nguyen and Tran [41] adopted warping-based triggers, which are more invisible for human inspection. Recently, [42] viewed the backdoor attack as a special multitask learning, where they leveraged adversarial perturbations to sample-agnostic. Cheng et al. [43] proposed to conduct the invisible attack in the feature space via style transfer. Different from previous works whose poisoned samples were generated in the pixel domain, [44] and [45] generated invisible trigger patterns in the frequency domain. Most recently, Li et al. [12] adopted DNN-based image steganography to generate invisible backdoor triggers. Compared with previous methods, this attack is not only invisible but can also bypass most existing backdoor defenses, since its trigger patterns are sample-specific.

Although a poisoned image is similar to its benign version in invisible attacks, however, its source label is usually different from the target label. In other words, all those methods are poison-label invisible attacks, where the poisoned samples seem to be mislabeled. Accordingly, an invisible attack still could be detected by examining the image-label relationship of training samples. To address this problem, a special subclass of invisible poisoning-based attacks, dubbed clean-label invisible attacks, was proposed. It is more serious and therefore worth more attention. Turner et al. [13] first explored the clean-label attack, where they leveraged adversarial perturbations or generative models to first modify some benign images from the target class and then conducted the standard invisible attack.

The modification process is to alleviate the effects of “robust features” contained in the poisoned samples to ensure that the trigger can be successfully learned by the DNNs. Recently, Zhao et al. [15] extended this idea in attacking video classification, where they adopted universal perturbation instead of a given one as the trigger pattern. Quiring and Rieck [46] proposed to conceal the trigger via image-scaling attacks [47]. Another interesting clean-label attack method is to inject the information of a poisoned sample generated by a previous visible attack into the texture of an image from the target class by minimizing their distance in the feature space, as suggested in [14]. Following the settings in [14], Souri et al. [48] formulated the backdoor attacks as a bilevel optimization [49], based on which they proposed a new clean-label backdoor attack. Most recently, Shumailov et al. [50] proposed to inject hidden backdoors via manipulating the order of training samples without changing samples.

In particular, clean-label backdoor attacks usually suffered from low attack effectiveness compared with poison-label attacks, although they are more stealthy. How to balance the stealthiness and effectiveness of attacks is still an open question and worth further exploration.

### 3) Optimized Backdoor Attacks:

Triggers are the core of poisoning-based attacks. As such, analyzing how to design a better trigger instead of simply using a given nonoptimized patch is of great significance and has attracted some attention. In general, backdoor attacks can be formulated as a bilevel optimization [49], i.e.,

$$\min_{w} R_s(D_t - D_s; w) + \lambda_1 \cdot R_b(D_s; t^*, w) + \lambda_2 \cdot R_p(D_t; t^*, w), \ s.t., \ t^* = \min \{ R_b(D_s; t, w) \}.$$  

Optimized attacks generated poisoned samples with optimized triggers to achieve better performance. To the best of our knowledge, Liu et al. [51] first explored this problem, where they proposed to optimize the trigger so that the important neurons can achieve the maximum values. After that, with the hypothesis that if a perturbation can induce most samples toward the decision boundary of the target class, it will serve as an effective trigger, [15], [37], and [52]...
proposed to generate a trigger through universal adversarial perturbation. These methods can be regarded as the heuristic solutions to the aforementioned bilevel optimization. Recently, [11], [39], [40], and [48] solved the bilevel optimization problem directly. For example, [11], [39], and [40] alternately optimized the upper level and lower level subproblems, while [48] adopted the gradient matching [53]. However, optimized backdoor attacks usually suffer from poor generalization, i.e., overfits to a particular model structure or model status. Although existing works introduced model-ensemble or carefully designed the alternately optimization process to alleviate the overfitting, how to better balance the effectiveness and generalization of the optimized triggers is still an important open question.

4) Semantic Backdoor Attacks: The majority of backdoor attacks, i.e., the nonsemantic attacks, assume that backdoor triggers are independent of benign images. As such, attackers need to modify the image in the digital space to activate hidden backdoors in the inference process. That raises the question if a semantic part of samples can also serve as the trigger pattern, such that the attacker is not required to modify the input at inference time to deceive the infected model. Bagdasaryan et al. first explored this problem and proposed a novel type of backdoor attacks [42], [54], i.e., the semantic backdoor attacks. Specifically, they demonstrated that assigning an attacker-chosen label to all images with certain features, e.g., green cars or cars with racing stripes, for training can create semantic backdoors in the infected DNNs. Accordingly, the infected model will automatically misclassify testing images containing predefined semantic information without any image modification. A similar idea was also explored in [55], where the hidden backdoor can be activated by the combination of certain objects in the image. Since these attacks do not require modifying images in the digital space, they are more malicious and worth further exploration.

5) Sample-Specific Backdoor Attacks: Currently, almost all backdoor attacks were sample agnostic, i.e., all poisoned samples contained the same trigger pattern. This property was widely used in the design of backdoor defenses, such as trigger synthesis-based defenses [25], [56]-[59] and saliency-based defenses [60], [61]. Nguyen and Tran proposed the first sample-specific backdoor attack, where different poisoned samples contain different trigger patterns. This attack bypassed many existing backdoor defenses for it broke their fundamental assumptions. However, it needs to control the training loss except for solely modifying training samples and their triggers are still visible, which significantly reduces its threat in real-world applications. After that, Li et al. proposed the first poison-only sample-specific backdoor attack with invisible trigger patterns, inspired by the advanced DNN-based image steganography. A similar idea is also explored in [63], where they embedded trigger patterns in the edge structure of poisoned images. Since these attacks can bypass most existing backdoor defenses, they pose a serious security threat and therefore worth further explorations.

6) Physical Backdoor Attacks: Different from previous digital attacks where attacks were conducted completely in the digital space, the physical space was also involved when generating poisoned samples in the physical attacks. Chen et al. [10] first explored the landscape of this attack, where they adopted a pair of glasses as the physical trigger to mislead the infected face recognition system developed in a camera. Further exploration of attacking face recognition in the physical world was also discussed by Wenger et al. [64]. A similar idea was also discussed in [7], where a post-it note was adopted as the trigger in attacking traffic sign recognition deployed in the camera. Recently, Li et al. [9] demonstrated that existing digital attacks fail in the physical world since the involved transformations (e.g., rotation, and shrinkage) change the location and appearance of triggers in attacked samples. This inconsistency will greatly reduce the performance of backdoor attacks. Based on this understanding, they proposed a transformation-based attack enhancement so that the enhanced attacks remain effective in the physical world. This attempt is an important step toward successful backdoor attacks in real-world applications.

7) All-to-All Backdoor Attacks: Based on the type of target labels, existing backdoor attacks can be divided into two main categories, including the all-to-one attacks and the all-to-all attacks. Specifically, all-to-one attacks assumed that all poisoned samples have the same target label no matter what their ground-truth label is, i.e., \( S(y) = y_t, \forall y \in \{1, \ldots, K\} \). In contrast, different poisoned samples may have different labels in all-to-all attacks. For example, the label shifting function was assigned as \( S(y) = (y + 1) \mod K \) in [7], [39], and [62]. The all-to-all attacks can bypass many target-oriented defenses (e.g., [25], [59], [61]) for their complicated target shifting, and therefore, are more serious compared with the all-to-one attacks. However, there were only a few studies on all-to-all attacks. How to better design the all-to-all attack and the analysis of its properties remain blank.

8) Black-Box Backdoor Attacks: Different from previous white-box attacks which required to access the training samples, black-box attacks adopted the settings that the training set is inaccessible. In practice, the training dataset is usually not shared due to privacy or copyright concerns; therefore, black-box attacks are more realistic than white-box ones. In general, black-box backdoor attackers generated some substitute training samples at first. For example, in [51], attackers generated some representative images of each class by optimizing images initialized from another dataset such that the prediction confidence of the selected class reaches maximum. With the substitute training samples, white-box attacks can be adopted for backdoor injection. Black-box backdoor attacks are significantly more difficult than white-box ones and there were only a few works in this area.

D. Attacks Against Other Fields or Paradigms
Currently, most existing backdoor attacks against other tasks or paradigms were still poisoning-based. Accordingly, except for task-specific requirements, most methods focused on: 1) how to design the trigger; 2) how to define the attack stealthiness; and 3) how to bypass potential defenses. The huge differences between different tasks and paradigms make the answers to the above-mentioned questions completely different. For example, the stealthiness in image-related tasks
can be defined as the pixelwise distance (e.g., $\ell^p$ norm) between the poisoned sample and its benign version; however, in natural language processing (NLP), changing even a word or character may still make the modification visible to human since it may cause grammar or spelling errors.

NLP is currently the most extensive research field in backdoor attacks besides image classification. Dai et al. [65] discussed how to attack against LSTM-based sentiment analysis. Specifically, they proposed a BadNets-like approach, where an emotionally neutral sentence was used as the trigger and was randomly inserted into some benign training samples. Chen et al. [66] further explored this problem, where three different types of triggers (i.e., char-level, word-level, and sentence-level triggers) were proposed and reached decent performance. Besides, Kurita et al. [16] demonstrated that sentiment classification, toxicity detection, and spam detection can also be attacked even after fine-tuning. Most recently, other backdoor attacks were also introduced, targeting different trigger types [67]–[70] and model components [71], [72] in different NLP tasks. Except for NLP-related tasks, researchers also revealed the backdoor threats in graph neural networks (GNN) [73]–[75], 3-D point cloud [76]–[78], semi-/self-supervised learning [79]–[81], reinforcement learning [82]–[84], model quantization [85]–[87], acoustics signal processing [88], [89], malware detection [90], [91], and other tasks [92]–[94].

Except for the classical training paradigm, how to backdoor collaborative learning, especially federated learning, has attracted the most attention. Bagdasaryan et al. [54] introduced the first backdoor attack against federated learning by amplifying the poisoned gradient of node servers. After that, Bhagoji et al. [95] discussed the stealthy model-poisoning backdoor attack, and Xie et al. [96] introduced a distributed backdoor attacks against the federated learning. Most recently, [97] theoretically verified that backdoor attacks are unavoidable if a model is vulnerable to adversarial examples under mild conditions in federated learning. Besides, the backdoor attacks toward meta federated learning [98] and feature-partitioned collaborative learning [99], [100] were also discussed. In contrast, some works [30], [101]–[103] also questioned whether federal learning is really easy to be attacked. Except for collaborative learning, the backdoor threat toward another important learning paradigm, i.e., transfer learning, was also discussed in [17], [104], and [105].

IV. NONPOISONING-BASED BACKDOOR ATTACKS

Except for poisoning-based backdoor attacks, the recent literature also proposed some nonpoisoning-based attacks. These methods embedded hidden backdoors not directly based on data poisoning during the training process. For example, attackers may directly change model weights or even the model structure without the training process. Their existence demonstrates that backdoor attacks could also happen at other stages (e.g., deployment stage) instead of simply the data collection or training stages, which further reveals the severity of backdoor threats.

A. Weights-Oriented Backdoor Attacks

In the weights-oriented backdoor attacks, attackers modified model parameters directly instead of through training with poisoned samples. To the best of our knowledge, Dumford and Scheirer [19] proposed the first weights-oriented attack where they adopted a greedy search across models with different perturbations applied to a pretrained model’s weights. It is also the first nonpoisoning-based backdoor attack. After that, Rakin et al. [20] introduced a bit-level weights-oriented backdoor attack, i.e., the targeted bit Trojan (TBT), which flipped critical bits of weights stored in the memory. The proposed method achieved remarkable performance, where attackers were able to mislead ResNet-18 [113] on the CIFAR-10 dataset [114] with 84 bit-flips out of 88 million weight bits. A similar idea was also introduced in [21], where attackers can significantly reduce the required flipping bits to embed hidden backdoors. Besides, Garg et al. [52] proposed to add adversarial perturbations on the model parameters of the benign model for injecting backdoors, showing a novel security threat of using publicly available trained models. Most recently, Zhang et al. [115] formulated the behavior of maintaining accuracy on benign samples as the consistency of infected models and provided a theoretical explanation of the adversarial weight perturbation (AWP) in backdoor attacks. Based on
the analysis, they also introduced a new AWP-based backdoor attack with better global and instancewise consistency.

Different from previous approaches where the backdoor is embedded in the parameters directly, Guo et al. [116] proposed TrojanNet to encode the backdoor in the infected DNNs activated through a secret weight permutation. Specifically, training a TrojanNet is similar to the multitask learning, although the benign task and malicious task share no common features. Besides, the authors also proved that the decision problem to determine whether the model contains a permutation that triggers the hidden backdoor is NP-complete, and therefore, the backdoor detection is almost impossible.

B. Structure-Modified Backdoor Attacks

Structure-modified backdoor attacks injected hidden backdoors into benign models by changing their model structures. These attacks could happen when using third-party models or in the deployment stage. To the best of our knowledge, Tang et al. [22] proposed the first structure-modified attack, where they inserted a trained malicious backdoor module (i.e., a sub-DNN) into the target model for embedding hidden backdoors. This attack was simple yet effective and the malicious module can be combined with all DNNs. A similar idea was also explored in [23], where attackers embedded malicious conditional logics into the target DNNs by adding malicious payload containing the conditional module and the trigger detector. Most recently, Qi et al. [24] proposed to directly replace instead of adding a narrow subnet of the benign model to conduct the backdoor attack. This method is effective in both digital and physical scenarios.

V. CONNECTION WITH RELATED REALMS

In this section, we discuss the similarities and differences between backdoor attacks and related realms. Those connections are summarized in Table III.

A. Backdoor Attacks and Adversarial Attacks

Both adversarial attacks and backdoor attacks modify the benign testing samples to make models misbehave during the inference process. Especially when the adversarial perturbations are sample-agnostic in universal adversarial attacks [38], [117], [118], these attacks seem to be the same. As such, researchers who are not familiar with the backdoor attack may question its research significance for it requires additional controls of the training process to some extent.

However, these attacks still have essential differences although they enjoy certain similarities: 1) From the aspect of the attacker’s capacity, adversarial attackers need to control the inference process (to a certain extent) but not the training process of models. Specifically, they need to query the model results or even gradients multiple times to generate adversarial perturbations by optimization given a fixed targeted model. In contrast, backdoor attackers require to modify some training stages (e.g., data collection and model training) without any additional requirements in the inference process. 2) From the perspective of attacked samples, the perturbation is known (i.e., nonoptimized) by backdoor attackers whereas adversarial attackers need to obtain it through the optimization process based on the output of the model. Such optimization in adversarial attacks requires multiple queries [160]–[162]. As such, adversarial attacks are unable to be real-time in many cases for the optimization process takes time. 3) Their mechanism also has essential differences. Adversarial vulnerability results from the differences in behaviors of models and humans. In contrast, backdoor attackers utilize the excessive learning ability of DNNs to build a latent connection between the trigger patterns and the target labels.

Most recently, there were also a few works studying the latent connection between adversarial attacks and backdoor attacks. For example, Weng et al. [163] empirically demonstrated that defending against adversarial attacks via adversarial training may increase the risks of backdoor attacks.

B. Backdoor Attacks and Data Poisoning

In general, there are two types of data poisoning, including the classical and the advanced one. The former one intends to reduce model generalization, i.e., letting the infected models behave well on training samples whereas having bad performance on testing samples. In contrast, advanced data poisoning makes infected models behave well on testing samples whereas having bad performance on some attacker-specified targeted samples which are not contained in the training set.

Data poisoning and (poisoning-based) backdoor attacks share many similarities in the training phase. In general, they all aim at misleading models in the inference process by introducing poisoned samples during the training process. However, they also have many intrinsic differences.
First, compared with classical data poisoning, backdoor attacks preserve the performance of predicting benign samples. In other words, backdoor attacks have different attacker goals compared with classical data poisoning. Besides, these attacks have different mechanisms. Specifically, the effectiveness of classical data poisoning is mostly due to the sensitivity of the training process so that even a small domain shift of training samples may lead to significantly different decision surfaces of infected models. Moreover, backdoor attacks are also more stealthy than classical data poisoning. Users can easily detect classical data poisoning by evaluating the performance of trained models on a local verification set, while this method has limited benefits in detecting backdoor attacks. Second, backdoor attacks are also different from advanced data poisoning. Specifically, there is no trigger in advanced data poisoning, which does not require modifying targeted samples during the inference process. Correspondingly, advanced data poisoning can only misclassify (a few) specific samples, which limits its threats in many scenarios.

In particular, the studies of existing data poisoning have also inspired the research on backdoor learning due to their similarities. For example, Hong et al. [144] demonstrated that the defense toward data poisoning may also have benefits in defending backdoor attacks, as illustrated in Section VI-A5.

VI. BACKDOOR DEFENSES

To alleviate the backdoor threats, several backdoor defenses were proposed. Existing methods mostly aim at defending against poisoning-based attacks and can be divided into two main categories, including empirical backdoor defenses and certified backdoor defenses. Specifically, empirical defenses were proposed based on some understandings of existing attacks and had decent performances in practice, whereas their effectiveness has no theoretical guarantee. In contrast, the validity of certified backdoor defenses is theoretically guaranteed under certain assumptions, whereas it is generally weaker than that of empirical defenses in practice. At present, certified defenses are all based on the random smoothing [164], while empirical ones have multiple types of approaches.

A. Empirical Backdoor Defenses

Intuitively, poisoning-based backdoor attacks are similar to unlocking a door with the corresponding key. In other words, there are three indispensable requirements to ensure the success of backdoor attacks, including: 1) having a hidden backdoor in the (infected) model; 2) containing triggers in (attacked) samples; and 3) the trigger and the backdoor are matched, as shown in Fig. 5. Accordingly, three main defense paradigms, including 1) trigger-backdoor mismatch; 2) backdoor elimination; and 3) trigger elimination, can be adopted to defend against existing attacks. Different types of defenses were proposed towards the aforementioned paradigms, which are summarized in Table IV and further illustrated as follows:

1) Preprocessing-Based Defenses: These methods introduce a preprocessing module before feeding samples into DNNs to change trigger patterns contained in attacked samples. As such, the modified triggers no longer match the hidden backdoor, therefore, preventing its activation.

Liu et al. [119] proposed the first preprocessing-based backdoor defense, where they adopted a pretrained auto-encoder as the preprocessor. Inspired by the idea that the trigger regions contributed most to the prediction, Doan et al. [120] introduced a two-stage image preprocessing approach (i.e., Februus). In the first stage, Februus used GradCAM [165] to identify influential regions, which will then be removed and replaced by a neutralized-color box. After that, Februus adopted a GAN-based inpainting method to reconstruct the masked regions to alleviate its adverse effects (e.g., BA drop). After that, Udeshi et al. [121] used the dominant color in the image to make a square-like trigger blocker in the preprocessing stage, which was adopted to locate and remove the backdoor trigger. This approach was motivated by the understanding that placing a trigger blocker at the position of trigger patterns in attacked images will significantly change model predictions. Villarreal-Vasquez and Bhargava [122] proposed to preprocess the image through style transfer. Recently, Li et al. [9] discussed the property of existing poisoning-based attacks with static trigger patterns. They demonstrated that if the appearance or location of the trigger is slightly changed, the attack performance may degrade sharply. Based on this observation, they proposed to adopt spatial transformations (e.g., shrinking, flipping) for the defense. Compared with previous methods, this method is more efficient since it requires almost no additional computational costs. A similar idea was explored in [123], where they introduced and evaluated more transformations in both fine-tuning and inference processes.

2) Model Reconstruction-Based Defenses: Different from preprocessing-based defenses, model reconstruction-based methods aim at removing hidden backdoors in the infected model by modifying suspicious models directly. As such, even if the trigger is contained in attacked samples, the reconstructed model will still correctly predict them since the hidden backdoors were already removed.

Liu et al. [119] proposed to retrain the trained suspicious model with some local benign samples to reduce backdoor threats. Its effectiveness is mostly due to the catastrophic forgetting of DNNs [166], i.e., the hidden backdoor is
gradually removed as the training goes since the retraining set contains no poisoned samples. This idea was further explored by Zeng et al. [129], where they formulated the retraining as a mini-max problem and adopted the implicit hypergradients to account for the interdependence between inner and outer optimization. Motivated by the observation that the backdoor-related neurons are usually dormant when predicting benign samples, Liu et al. [124] proposed to prune those neurons to remove the hidden backdoor. Specifically, they proposed a fine-pruning method, which first prunes the DNNs and then fine-tunes the pruned network to combine the benefits of the pruning and fine-tuning defenses. A similar idea was further explored in [128], where they used adversarial weight perturbation (AWP) to amplify the differences between benign and malicious neurons. Zhao et al. [125] showed that the hidden backdoor of infected DNNs can be repaired based on the inner and outer optimization. Motivated by the observation that the distillation process perturbs backdoor-related neurons, and therefore, can remove hidden backdoors.

3) Trigger Synthesis-Based Defenses: Except for eliminating hidden backdoors directly, trigger synthesis-based defenses first synthesize the backdoor trigger, followed by the second stage where hidden backdoors are removed by suppressing trigger effects. These defenses enjoy certain similarities with reconstruction-based ones in the second stage. For example, pruning and retraining are the common techniques used in removing the hidden backdoor in both types of defenses. However, compared with the reconstruction-based defenses, the trigger information obtained in synthesis-based defenses makes the removal process more effective and efficient.

To the best of our knowledge, Wang et al. [25] proposed the first trigger synthesis-based defense (i.e., neural cleanse), where defenders first obtained potential trigger patterns toward every class and then determined the final synthetic trigger and its target label based on anomaly detection. A similar idea was also discussed in [56], [134], [135], and [142], where they designed different trigger reversion or detection techniques. Qiao et al. [130] noticed that the reversed trigger synthesized by neural cleanse is usually significantly different from that was used in the training process, inspired by which they first discussed the generalization of the backdoor trigger. They demonstrated that infected models will generalize their original triggers during the training process. Accordingly, they proposed to recover the trigger distribution, rather than a specific trigger for the defense, based on a max-entropy staircase approximator. A similar idea was also discussed in [131], where they proposed a GAN-based method to synthesize trigger distribution. In [57], they showed that the detection process used for determining the synthetic trigger in [25] suffers from several failure modes, based on which they proposed a new defense method. Besides, Cheng et al. [132] revealed that the $\ell^\infty$ norm of the activation values can be used to distinguish backdoor-related neurons based on the synthetic trigger. Accordingly, they proposed to perform $\ell^\infty$-based neuron pruning to remove neurons with high activation values in response to the trigger. Similarly, Aiken et al. [133] also proposed to remove the hidden backdoor by pruning DNNs based on the synthetic trigger from another perspective. Moreover, Shen et al. [58] proposed an efficient trigger synthesis-based defense. Different from previous defenses, which needed to generate all potential triggers toward each class, this defense selects only one class for trigger optimization in each round, inspired by the K-Arm bandit [169]. Recently, Hu et al. [137] designed a topological prior to improve trigger synthesis. Note that all previous defenses are white-box, requiring defenders have the access to model source files. Most recently, a few black-box synthesis-based defenses [59], [136] were also proposed, where defenders can reverse trigger patterns even when they can only obtain model predictions (e.g., probability vectors or predicted labels).

4) Model-Diagnosis-Based Defenses: These defenses justify whether a suspicious model is infected based on a pre-trained meta-classifier and refuse to deploy infected models. Since only the benign models are used for deployment, it naturally eliminates the hidden backdoor.

| Defense Paradigm       | Defense Sub-category                       | Literature                                                                 |
|------------------------|--------------------------------------------|-----------------------------------------------------------------------------|
| Trigger-backdoor Mismatch | Preprocessing-based Defenses               | [119], [120], [121], [122]                                                 |
|                        |                                            | [9], [123]                                                                  |
| Backdoor Elimination   | Model Reconstruction based Defenses         | [119], [124], [125], [126]                                                 |
|                        |                                            | [127], [128], [129]                                                        |
|                        | Trigger Synthesis based Defenses           | [125], [126], [130], [131], [132], [133], [134]                            |
|                        |                                            | [135], [58], [59], [136]                                                   |
|                        | Model Diagnosis based Defenses             | [140], [141], [143], [139]                                                 |
|                        |                                            | [140], [141], [142]                                                        |
|                        | Poison Suppression based Defenses          | [143], [144], [145], [146]                                                 |
|                        |                                            | [27], [147]                                                                |
|                        | Training Sample Filtering based Defenses   | [143], [149], [150], [151]                                                 |
|                        |                                            | [152], [61], [153], [154]                                                  |
|                        |                                            | [155]                                                                      |
| Trigger Elimination    | Testing Sample Filtering based Defenses     | [156], [157], [143], [158]                                                 |
|                        |                                            | [159]                                                                      |
To the best of our knowledge, Kolouri et al. [26] first discussed how to diagnose a given model. Specifically, they jointly optimized some universal litmus patterns (ULPs) and a meta-classifier, which was further used to diagnose suspicious models based on the predictions of obtained ULPs. Different from the previous defense where both infected models and benign models are required to train the meta-classifier, an effective meta-classifier can be trained only on benign models based on the strategies proposed in [138]. Besides, motivated by the observation that the heatmaps from benign and infected models have different characteristics, Huang et al. [60] adopted an outlier detector as the meta-classifier based on three extracted features of generated saliency maps. In [139], they designed a one-pixel signature representation, based on which to distinguish benign and infected models. Besides, Wang et al. [140] discussed how to detect whether a given mode is benign or infected in the data-limited and data-free cases. Most recently, [141] revealed that benign models and infected DNNs have significant topologically structural differences, which can be used to diagnose suspicious models.

5) Poison Suppression-Based Defenses: These defenses depress the effectiveness of poisoned samples during the training process to prevent the creation of hidden backdoors. Du et al. [143] first explored poison suppression-based defenses, where they adopted noisy SGD to learn differentially private DNNs for the defense. With the randomness in the training process, the malicious effects of poisoned samples were reduced by random noise, preventing backdoor creation. Motivated by the observation that the $\ell^2$ norm of the gradients of poisoned samples has significantly higher magnitudes than those of benign samples and their gradient orientations are also different, Hong et al. [144] adopted differentially private stochastic gradient descent (DPSGD) to clip and perturb individual gradients during the training process. Accordingly, the trained model had no hidden backdoor as well as its robustness toward targeted adversarial attacks was also increased. Besides, Borgain et al. [145] revealed that introducing strong data augmentation methods (e.g., CutMix [170]) can effectively prevent the creation of hidden backdoors for they significantly perturbed the trigger patterns during the training process. Li et al. [27] proposed a gradient ascent-based antibackdoor method, based on the observations that backdoor attacks have faster learning on poisoned data and target-class dependency. Most recently, Huang et al. [147] revealed that the hidden backdoors are learned mostly due to the end-to-end supervised training paradigm, based on which they proposed a simple yet effective decoupling-based training method for backdoor suppression.

6) Training Sample Filtering-Based Defenses: These defenses aim at filtering poisoned samples from the training dataset. After the filtering process, only benign samples or purified poisoned samples will be used in the training process, which eliminates backdoor creation from the source. To the best of our knowledge, Tran et al. [148] first explored how to filter malicious samples from the training set. Specifically, they demonstrated that poisoned samples tend to leave behind a detectable trace in the spectrum of the covariance of feature representations, which can be used to filter poisoned samples from the training set. Recently, Hayase and Kong [153] introduced robust covariance estimation to amplify the spectral signature of poisoned samples, based on which they designed a more effective filtering method (i.e., SPECTRE). Also, inspired by the idea that poisoned samples and benign samples should have different characteristics in the hidden feature space, Chen et al. [149] proposed a two-stage filtering method, including: 1) clustering the activations of training samples in each class into two clusters and 2) determining which, if any, of the clusters corresponds to poisoned samples. A similar idea was also explored in [151]. However, Tang et al. [150] demonstrated that simple target contamination can cause the representation of poisoned samples to be less distinguishable from that of benign ones, and therefore, most existing filtering-based defenses can be easily bypassed. To address this problem, they proposed a more robust sample filter, based on representation decomposition and its statistical analysis. Different from previous methods, Chan and Ong [152] separated poisoned samples based on signals contained in input gradients. A similar idea was explored in [61], where they adopted the saliency map to identify trigger regions and filter poisoned samples. Besides, Wang et al. [154] formulated the filtering as optimal data selection, based on which they proposed a unified framework to filter different types of malicious training samples. Most recently, Zeng et al. [155] revealed that poisoned samples of existing attacks had some high-frequency artifacts even if their trigger patterns are invisible in the input space. Based on this observation, they designed a simple yet effective filtering method based on those artifacts.

7) Testing Sample Filtering-Based Defenses: These defenses also filter malicious samples, whereas the filtering happened in the inference instead of the training process. Only benign testing or purified attacked samples will be predicted by the deployed model. These defenses prevent backdoor activation for they can remove trigger patterns. Motivated by the observation that most of the existing backdoor triggers are input-agnostic, Gao et al. [156] proposed to filter attacked samples via superimposing various image patterns on the suspicious samples. The smaller the randomness among the predictions of perturbed inputs, the higher the probability that the suspicious sample is attacked. Subedar et al. [157] adopted model uncertainty to distinguish between benign and attacked samples. After that, Du et al. [143] treated the filtering as outlier detection, based on which they proposed a differential privacy-based filtering method. Besides, Jin et al. [158] proposed to detect attacked samples based on existing detection-based adversarial defenses [171]–[173]. Most recently, a lightweight method was proposed in [159], which can filter attacked samples without labeled samples or prior assumptions on trigger patterns.

B. Certified Backdoor Defenses

Although multiple empirical defenses have been proposed and reached decent performance against some backdoor attacks, almost all of them were bypassed by the following adaptive attacks [14], [188]. To terminate this “cat-and-mouse
chasing game," Wang et al. [28] took the first step toward the certified defense against backdoor attacks based on the random smoothing technique [164]. Randomized smoothing was originally developed to certify robustness against adversarial examples, where the smoothed function was built from the base function via adding random noise to the data vector to certify the robustness of a classifier under certain conditions. Similar to [189], Wang et al. [28] treated the entire training procedure of the classifier as the base function to generalize classical randomized smoothing to defend against backdoor attacks. Weber et al. [29] demonstrated that directly applying randomized smoothing, as in [28], will not provide high certified robustness bounds. Instead, they proposed a unified framework with the examination of different smoothing noise distributions and provided a tightness analysis for the robustness bound. Most recently, a few studies [190]–[192] also adopted ensemble techniques (e.g., Bagging [193]) in designing certified defenses to further improve effectiveness.

C. Evaluation Metrics

1) Metrics for Detection-Like Empirical Defenses: Model-diagnosis-based defenses and testing sample filtering-based defenses are all detection-like methods, whose main target is to identify whether a suspicious object (e.g., a trained DNN or sample) is malicious. This is essentially a binary classification problem. To evaluate their performance, three metrics [194], including: 1) precision; 2) recall; and 3) F1-score, are usually adopted. The higher the precision, recall, and F1 score, the better the defense performance.

2) Metrics for Nondetection-Like Empirical Defenses: Except for model diagnosis-based and testing sample filtering-based defenses, all other methods are nondetection-like. Their main target is to achieve correct predictions for both benign and attacked samples. Accordingly, both BA and ASR (as defined in Section II-A) are also adopted for the evaluation. In particular, although a detection process is also involved in training sample filtering-based defenses, three metrics (i.e., precision, recall, and F1-score) described earlier are not suitable for their evaluation. These defenses may try to discard as many poisoned samples as possible to reduce the possibility of creating hidden backdoors trained on the filtered dataset, even with the sacrifice of certain benign samples.

3) Metrics for Certified Defenses: As mentioned in Section VI-B, existing certified backdoor defenses all adopted random smoothing. As such, these methods can provide a certified radius, where all perturbation within the $\ell^p$ ball with the certified radius cannot change the prediction of the model under certain assumptions. To evaluate their performance, people usually use the: 1) benign accuracy; 2) certified rate; and 3) certified accuracy as the evaluation metric [28], [29]. Specifically, the benign accuracy indicates how well the (smoothed) classifier performs in classifying benign samples; the certified rate is the fraction of samples that can be certified at radius greater than the certified radius; and the certified accuracy is the fraction of the test set which is classified correctly and is certified as robust with a radius greater than the certified radius. The higher the benign accuracy, certified rate, and certified accuracy, the better the defense performance.

VII. Benchmark Datasets

Similar to that adversarial learning and data poisoning, most of the existing backdoor-related literature focused on image classification tasks. In this section, we summarize all classical image classification benchmark datasets in Table V.

Specifically, these benchmark datasets can be divided into three main categories, including natural image recognition, traffic sign recognition, and face recognition. The former ones are classical in the image classification, while the second and third ones are tasks requiring strict security guarantees. We recommend that future studies should be evaluate these datasets to facilitate comparison and ensure fairness.

VIII. Outlook of Future Directions

As presented earlier, many works have been proposed in the literature on backdoor learning, covering several branches and different scenarios. However, we believe that the development of this field is still in its infancy, as many critical problems of backdoor learning have not been well studied. In this section, we present five potential research directions to inspire the future development of backdoor learning.

A. Trigger Design

The effectiveness and efficiency of poisoning-based backdoor attacks are closely related to their trigger patterns. However, the trigger of most existing attacks was designed in a heuristic (e.g., design with universal perturbation) or even a nonoptimized way. How to better optimize the trigger pattern (e.g., based on bilevel optimization) is still an important open question. Besides, only the effectiveness and trigger invisibility were considered in the trigger design. Other criteria, such as minimal poisoning rate and trigger generalization, are also worth further exploration.

B. Semantic and Physical Backdoor Attacks

As presented in Section III-C, semantic and physical attacks are more serious threats to AI systems in practical scenarios, while their studies are left far behind compared with other types of backdoor attacks. More thorough studies to obtain better understanding of these attacks would be important steps toward alleviating the backdoor threats in practice. For example, one may explore whether other physical phenomena (e.g., specific illumination), can also serve as effective physical trigger patterns and why semantic triggers are also effective.

C. Attacks Toward Other Tasks

The success of backdoor attacks heavily relied on the trigger design according to the characteristics of the target task. For example, the visual invisibility of the trigger is one of the critical criteria in visual tasks, which ensures stealthiness. However, the design of backdoor triggers in different tasks could be quite different (e.g., hiding a trigger into a sentence when attacking NLP-related tasks is quite different from hiding a trigger into an image). As such, it is necessary to study task-specific backdoor attacks. Currently, existing backdoor attacks mainly focused on computer vision tasks,
especially image classification. The research on other tasks (e.g., recommendation system, speech recognition, and NLP) have not been well studied. Besides, regression as another important paradigm deserves more attention and backdoor-related exploration.

D. Effective and Efficient Defenses

Although many types of empirical backdoor defenses have been proposed (as demonstrated in Section VI), almost all of them can be bypassed by subsequent adaptive attacks. Besides, except for the preprocessing-based defenses, existing defenses usually suffer from high computational costs. More efforts in designing effective and efficient defenses (e.g., analyzing the weaknesses of existing attacks and how to reduce the computational costs of defenses) should be made to keep up with the fast pace of backdoor attacks. For example, it would be possible to locate the trigger patterns or malicious hidden features based on explainable AI (XAI) methods. Besides, how to design black-box defenses is also worth more attention since these methods are more practical in reality. Moreover, certified backdoor defenses are important yet currently have been rarely studied, which deserve more explorations.

E. Mechanism Exploration

The principle of backdoor generation and the activation mechanism of backdoor triggers are the Holy Grail problems in backdoor learning. For example, why hidden backdoors can be created and what happens inside the infected models when the trigger appears have not been carefully studied in existing works. A deeper understanding of their intrinsic mechanisms can guide the design of more effective attacks and defenses, and the understanding of DNNs.
IX. CONCLUSION

Backdoor learning, including backdoor attacks and backdoor defenses, is a critical and booming research area. In this survey, we summarized and categorized existing backdoor attacks and proposed a unified framework for analyzing poisoning-based backdoor attacks. Specifically, we proposed seven different criteria for classifying existing attacks. Among all these attacks, we believed that semantic, sample-specific, and black-box poisoning-based backdoor attacks are worth more attention since they are more threatening in practice. We also discussed the relation between backdoor attacks and related research areas (i.e., data poisoning and adversarial attacks) and analyzed existing defenses. Specifically, we identified three fundamental defense paradigms and divided existing defenses into eight main categories. Classical benchmark datasets and potential research directions were illustrated at the end. Note that almost all studies in this field were completed in the last four years and the cat-and-mouse game between attacks and defenses is likely to continue in the future. We hope that this survey could remind researchers of backdoor threats and provide a timely view. It would be an important step toward building more robust and secure DNNs.

ACKNOWLEDGMENT

The authors sincerely thank Dr. Baoyuan Wu from The Chinese University of Hong Kong (Shenzhen), Shenzhen, Guangdong, China, and Dr. Bo Li from the University of Illinois Urbana–Champaign, Urbana, IL, USA, for their helpful comments and suggestions on an early draft of this article. This work was partly done when Yiming Li was a research intern at Tencent AI Laboratory, Shenzhen.

REFERENCES

[1] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” in Proc. ICLR, 2015.
[2] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” in Proc. ICLR, 2018.
[3] J. Bai et al., “Targeted attack for deep hashing based retrieval,” in Proc. ECCV, 2020, pp. 618–634.
[4] D. Wu, S. Xiang, Y. Wang, “Adversarial weight perturbation helps robust generalization,” in Proc. NeurIPS, 2020, pp. 2958–2969.
[5] Y. Bai, Y. Zeng, Y. Jiang, S.-T. Xie, X. Ma, and Y. Wang, “Improving adversarial robustness via channel-wise activation suppressing,” in Proc. ICLR, 2021.
[6] Y. Li et al., “Semi-supervised robust training with generalized perturbed neighborhood,” Pattern Recognit., vol. 124, Apr. 2022, Art. no. 108472.
[7] T. Gu, K. Liu, B. Doan-Gavitt, and S. Garg, “BadNets: Evaluating backdooring attacks on deep neural networks,” IEEE Access, vol. 7, pp. 47230–47244, 2019.
[8] Y. Liu, X. Ma, J. Bailey, and F. Lu, “Reflection backdoor: A natural backdoor attack on deep neural networks,” in Proc. ECCV, 2020, pp. 182–199.
[9] Y. Li, T. Zhai, Y. Jiang, Z. Li, and S.-T. Xie, “Backdoor attack in the physical world,” in Proc. ICLR Workshop, 2021.
[10] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, “Targeted backdoor attacks on deep learning systems using data poisoning,” 2017, arXiv:1712.05526.
[11] S. Li, M. Xue, B. Zhao, H. Zhu, and X. Zhang, “Invisible backdoor attacks on deep neural networks via steganography and regularization,” IEEE Trans. Dependable Secure Comput., vol. 18, no. 5, pp. 2088–2105, Sep./Oct. 2021.
[12] Y. Li, Y. Li, B. Wu, L. Li, R. He, and S. Lyu, “Invisible backdoor attack with sample-specific triggers,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 16463–16472.
[13] A. Turner, D. Tsipras, and A. Madry, “Label-consistent backdoor attacks,” 2019, arXiv:1912.02771.
[14] A. Saha, A. Subramanya, and H. Pirsiavash, “Hidden trigger backdoor attacks,” in Proc. AAAI, 2020, pp. 11957–11965.
[15] S. Zhao, X. Ma, X. Zheng, J. Bailey, J. Chen, and Y.-Q. Jiang, “Eco-backdoor label attacks on video recognition models,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 14443–14452.
[16] K. Kurita, P. Michel, and G. Neubig, “Weight poisoning attacks on pre-trained models,” in Proc. ACL, 2020, pp. 2793–2806.
[17] S. Wang, S. Nepal, C. Rudolph, M. Grobler, S. Chen, and T. Chen, “Backdoor attacks against transfer learning with pre-trained deep learning models,” IEEE Trans. Services Comput., early access, Jun. 9, 2020, doi: 10.1109/TSC.2020.3000900.
[18] Y. Ge et al., “Anti-distillation backdoor attacks: Backdoors can really survive in knowledge distillation,” in Proc. ACM MM, 2021, pp. 826–834.
[19] J. Dumford and W. Scheirer, “Backdooroing convolutional neural networks via targeted weight perturbations,” 2018, arXiv:1812.05128.
[20] A. S. Rakin, Z. He, and D. Fan, “TBT: Targeted neural network attack with bit trojan,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 13198–13207.
[21] H. Chen, C. Fu, J. Zhao, and F. Koushanfar, “ProFlip: Targeted trojan attack with progressive bit flips,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 7178–7177.
[22] R. Tang, M. Du, N. Liu, F. Yang, and X. Hu, “An embarrassingly simple approach for trojan attack in deep neural networks,” in Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2020, pp. 218–228.
[23] Y. Li, J. Hu, H. Wang, C. Chen, and Y. Liu, “DeepPayload: Black-box backdoor attack on deep learning models through neural payload injection,” in Proc. IEEE/ACM 43rd Int. Conf. Softw. Eng. (ICSE), May 2021, pp. 263–274.
[24] X. Qi, J. Zhu, C. Xie, and Y. Yang, “Subnet Replacement: Deployment-stage backdoor attack against deep neural networks in gray-box setting,” in Proc. ICLR Workshop, 2021.
[25] B. Wang et al., “Neural cleanse: Identifying and mitigating backdoor attacks in deep neural networks,” in Proc. IEEE Symp. Secur. Privacy (SP), May 2019, pp. 707–723.
[26] S. Kolouri, A. Saha, H. Pirsiavash, and H. Hoffmann, “Universal litmus patterns: Revealing backdoor attacks in CNNs,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 301–310.
[27] Y. Li, X. Lyu, N. Koren, L. Lyu, B. Li, and X. Ma, “Anti-backdoor learning: Training clean models on poisoned data,” in Proc. NeurIPS, 2021, pp. 14900–14912.
[28] B. Wang et al., “On certifying robustness against backdoor attacks via randomized smoothing,” in Proc. CVPR Workshop, 2020.
[29] M. Weber, X. Xu, B. Karlaš, C. Zhang, and B. Li, “RAB: Provably robust against backdoor attacks,” 2020, arXiv:2003.08904.
[30] C. Xie, M. Chen, P.-Y. Chen, and B. Li, “Crlf: Certifiably robust federated learning against backdoor attacks,” in Proc. ICML, 2021, pp. 11372–11382.
[31] Y. Liu et al., “A survey on neural trojans,” in Proc. 21st Int. Symp. Quality Electron. Design (ISQED), Mar. 2020, pp. 33–39.
[32] M. Goldblum et al., “Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses,” IEEE Trans. Pattern Anal. Mach. Intell., early access, May 25, 2022, doi: 10.1109/TPAMI.2022.3162397.
[33] S. Kaviani and I. Sohn, “Defense against neural trojan attacks: A survey,” Neurocomputing, vol. 423, pp. 651–667, Jan. 2021.
[34] Y. Gao et al., “Backdoor attacks and countermeasures on deep learning: A comprehensive review,” 2020, arXiv:2007.10760.
[35] S. Li, S. Ma, M. Xue, and B. Zi Hao Zhao, “Deep learning backdoors,” 2020, arXiv:2007.08273.
[36] W. Guo, B. Tondi, and M. Barni, “An overview of backdoor attacks against deep neural networks and possible defenses,” 2021, arXiv:2111.08429.
[37] H. Zhong, C. Liao, A. C. Squicciarini, S. Zhu, and D. Miller, “Backdoor embedding in convolutional neural network models via invisible perturbation,” in Proc. 10th ACM Conf. Data Appl. Secur. Privacy, Mar. 2020, pp. 97–108.
[38] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1765–1773.

[39] K. Doan, Y. Lao, W. Zhao, and P. Li, "LIRA: Learnable, imperceptible and robust backdoor attacks," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 11966–11976.

[40] K. Doan, Y. Lao, and P. Li, "Backdoor attack with imperceptible input data and latent modification," in Proc. NeurIPS, 2021, pp. 18944–18957.

[41] T. A. Nguyen and A. T. Tran, "WaNet: Imperceptible warping-based backdoor attack," in Proc. ICLR, 2021.

[42] E. Bagdasaryan and V. Shmatikov, "Blind backdoors in deep learning models," in Proc. USENIX Secur., 2021, pp. 1505–1521.

[43] S. Cheng, Y. Liu, S. Ma, and X. Zhang, "Deep feature space trojan attack of neural networks by controlled detoxification," in Proc. AAAI, 2021, pp. 1148–1156.

[44] H. Abed Al Kader Hammoud and B. Ghanem, "Check your other works," in Proc. Annu. Comput. Secur. Appl. Conf., Dec. 2021, pp. 2048–2058.

[45] L. Li, D. Song, X. Li, J. Zeng, R. Ma, and X. Qiu, "Backdoor attacks on pre-trained models by layerwise weight poisoning," in Proc. Conf. Empirical Methods Natural Lang. Process., 2021, pp. 3023–3032.

[46] Z. Zhang, J. Jia, B. Wang, and N. Z. Gong, "Backdoor attacks to graph neural networks," in Proc. NeurIPS Workshop, 2020, pp. 15–26.

[47] Z. Xi, R. Pang, S. Ji, and T. Wang, "Graph backdoor," in USENIX Secur., 2021, pp. 1523–1540.

[48] J. Chen, H. Xiong, H. Zhang, J. Jiang, G. Jiang, and Y. Liu, "Dyn-backdoor: Backdoor attack on dynamic link prediction," 2021, arXiv:2109.05507.

[49] G. Tian, W. Jiang, W. Liu, and Y. Mu, "Poisoning MorphNet for clean-label backdoor attack to point clouds," 2021, arXiv:2105.04839.

[50] Z. Xiang, D. J. Miller, S. Chen, X. Li, and G. Kesidis, "A backdoor attack against 3D point cloud classifiers," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 7591–7607.

[51] X. Li et al., "PointBA: Towards backdoor attacks in 3D point cloud," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 16492–16501.

[52] Z. Yan, J. Wu, G. Li, S. Li, and M. Guizani, "Deep neural backdoor in semi-supervised learning: Threats and countermeasures," IEEE Trans. Inf. Forensics Security, vol. 16, pp. 4827–4842, 2021.

[53] J. Jia, Y. Liu, and N. Z. Gong, "Badencoder: Backdoor attacks to pre-trained encoders in self-supervised learning," in Proc. IEEE S&P, 2022.

[54] N. Carlini and A. Terzis, "Poisoning and backdooring contrastive learning," in Proc. ICLR, 2022.

[55] P. Kiourti, K. Wardega, S. Jha, and W. Li, "TroDRL: Evaluation of backdoor attacks on reinforcement learning," in Proc. 57th ACM IEEE Design Autom. Conf. (DAC), Jul. 2020, pp. 1–6.

[56] L. Fang, Z. Javed, X. Wu, W. Guo, X. Xing, and D. Song, "BACKDOORL: Backdoor attack against competitive reinforcement learning," in Proc. 13th Int. Joint Conf. Artif. Intell., Aug. 2021, pp. 3699–3705.

[57] C. Ashcraft and K. Karra, "Poisoning deep reinforcement learning agents with in-distribution triggers," in Proc. ICLR Workshop, 2021.

[58] H. Ma et al., "Quantization backdoors to deep learning commercial frameworks," 2021, arXiv:2108.09187.

[59] S. Hong, M.-A. Panaite-scu-Liess, Y. Kaya, and T. Dumitrescu, "QuANTI-zation: Exploiting quantization artifacts for achieving adversarial outcomes," in Proc. NeurIPS, 2021, pp. 9303–9316.

[60] X. Pan, M. Zhang, Y. Yan, and M. Yang, "Understanding the threats of trojaned quantized neural network in model supply chains," in Proc. Annu. Comput. Secur. Appl. Conf., Dec. 2021, pp. 634–645.

[61] T. Zhai, Y. Li, Z. Zhang, B. Wu, Y. Jiang, and S.-T. Xia, "Backdoor attack against speaker verification," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Jun. 2021, pp. 2560–2564.

[62] S. Koffas, J. Xu, M. Conti, and S. Picek, "Can you hear it? Backdoors attacks via ultrasonic triggers," 2021, arXiv:2107.14569.

[63] C. Li et al., "Backdoor attack on machine learning based Android malware detectors," IEEE Trans. Dependable Secure Comput., 2021, DOI: 10.1109/TDSC.2021.3094824.

[64] S. Severi, J. Meyer, S. Coull, and A. Oprea, "Explanation-guided backdoor poisoning attacks against malware classifiers," in Proc. USENIX Secur. Conf., 2021, pp. 1487–1504.

[65] Y. Li, Y. Li, Y. Lv, Y. Jiang, and S.-T. Xia, "Hidden backdoor attack against semantic segmentation models," in Proc. ICLR Workshop, 2021.

[66] S. Fang and A. Choromanska, "Backdoor attacks on the DNN interpretation system," in Proc. AAAI, 2021.

[67] Y. Li, H. Zhong, X. Ma, Y. Jiang, and S.-T. Xia, "Few-shot backdoor attacks on visual object tracking," in Proc. ICLR, 2022.
Yong Jiang received the M.S. and Ph.D. degrees in computer science from Tsinghua University, Beijing, China, in 1998 and 2002, respectively. Since 2002, he has been with the Tsinghua Shenzhen International Graduate School, Tsinghua University, where he is currently a Full Professor. His researches have been published in multiple top-tier journals and conferences, including the IEEE TRANSACTIONS ON COMPUTERS (ToC), IEEE TRANSACTIONS ON MULTIMEDIA (TMM), IEEE TRANSACTIONS ON SIGNAL PROCESSING (TSP), IEEE Conference on Computer Vision and Pattern Recognition (CVPR), International Conference on Learning Representations (ICLR), and European Conference on Computer Vision (ECCV). His research interests include computer vision, machine learning, Internet architecture, and its protocols, and IP routing technology. Dr. Jiang has received several best paper awards, such as IWQoS 2018.

Zhifeng Li (Senior Member, IEEE) received the Ph.D. degree from The Chinese University of Hong Kong, Hong Kong, China, in 2006. He was a Post-Doctoral Fellow with The Chinese University of Hong Kong and Michigan State University, East Lansing, MI, USA, for several years. He was a Full Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen. He is currently a Top-Tier Principal Researcher with Tencent Data Platform, Shenzhen, China. His research interests include face recognition, computer vision, deep learning, and pattern recognition.

Dr. Li is a fellow of the British Computer Society. He is currently serving on the Editorial Board of Pattern Recognition, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and Neurocomputing.

Shu-Tao Xia received the B.S. degree in mathematics and the Ph.D. degree in applied mathematics from Nankai University, Tianjin, China, in 1992 and 1997, respectively. From 1997 to 1999, he was with the Research Group of Information Theory, Department of Mathematics, Nankai University, Tianjin, China. From 1997 to 1998 and in 1998, he visited the Department of Information Engineering, The Chinese University of Hong Kong, Shenzhen, Guangdong, China. Since 2004, he has been with the Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, where he is currently a Full Professor. His research has been published in multiple top-tier journals and conferences, including the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (TPAMI), IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), International Conference on Computer Vision (ICCV), and International Conference on Learning Representations (ICLR). His current research interests include coding and information theory, machine learning, and deep learning.