Backdoor Attack and Defense on Deep Learning: A Survey

Yang Bai, Gaojie Xing, Hongyan Wu, Zhihong Rao, Chuan Ma, Shiping Wang, Xiaolei Liu, Yimin Zhou, Jiajia Tang, Kaijun Huang, and Jiale Kang

***Abstract*—Deep learning, as an important branch of machine learning, has been widely applied in computer vision, natural language processing, speech recognition, and more. However, recent studies have revealed that deep learning systems are vulnerable to backdoor attacks. Backdoor attackers inject a hidden backdoor into the deep learning model, such that the predictions of the infected model will be maliciously changed if the hidden backdoor is activated by input with a backdoor trigger while behaving normally on any benign sample. This kind of attack can potentially result in severe consequences in the real world. Therefore, research on defending against backdoor attacks has emerged rapidly. In this article, we have provided a comprehensive survey of backdoor attacks, detections, and defenses previously demonstrated on deep learning. We have investigated widely used model architectures, benchmark datasets, and metrics in backdoor research and have classified attacks, detections and defenses based on different criteria. Furthermore, we have analyzed some limitations in existing methods and, based on this, pointed out several promising future research directions. Through this survey, beginners can gain a preliminary understanding of backdoor attacks and defenses. Furthermore, we anticipate that this work will provide new perspectives and inspire extra research into the backdoor attack and defense methods in deep learning.**

Received 22 November 2023; revised 24 July 2024, 10 September 2024, and 7 October 2024; accepted 15 October 2024. This work was supported in part by Sichuan Science and Technology Department Key Research and Development Program under Grant 2024YFG0003, in part by Sichuan Science and Technology Program under Grant 2024ZDZX0007, in part by Zhejiang Lab OpenResearch Project under Grant K2022PDOAB06, in part by the Open Fund of Advanced Cryptography and System Security Key Laboratory of Sichuan Province under Grant SKLACSS-202403, and in part by Chengdu University of Information Technology 2022 Annual Research Initiation Project under Grant KYTZ2022107. *(Corresponding author: Gaojie Xing.)*

Yang Bai, Gaojie Xing, Hongyan Wu, Yimin Zhou, Jiajia Tang, Kaijun Huang, and Jiale Kang are with the School of Cybersecurity (Xin Gu Industrial College), and SUGON Industrial Control and Security Center, Chengdu University of Information Technology, Chengdu 610225, China (e-mail: alicepub@163.com; restartxgj@163.com; wuhy0126@163.com; yiminzhou@cuit.edu.cn; zhuo_2024@163.com; huangkaijun_cuit@163.com; kang131@foxmail.com).

Zhihong Rao is with the 30th Institute of Electronics Technology Group, Chengdu 610041, China (e-mail: zhongrao2024@163.com).

Chuan Ma is with the School of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: chuan.ma@cqu.edu.cn).

Shiping Wang is with the College of Computer and Data Science and the Fujian Provincial Key Laboratory of Network Computing and Intelligent Information Processing, Fuzhou University, Fuzhou 350108, China (e-mail: shipingwangphd@163.com).

Xiaolei Liu is with the Institute of Computer Application, China Academy of Engineering Physics, Mianyang 621900, China (e-mail: luxaole@gmail.com).

Digital Object Identifier 10.1109/TCSS.2024.3482723

***Index Terms*—Backdoor attacks, backdoor defenses, backdoor detections, deep learning.**

I. Introduction

DEEP learning, an important branch of machine learning, employs algorithms that use multiple processing layers with complex structures or various nonlinear transformations to achieve high-level abstractions of data [1]. It has been extensively researched in various domains, such as computer vision [2], natural language processing [3], and speech recognition [4]. Furthermore, the use of deep learning to ensure the security of internet applications and data has become ubiquitous in real life. For example, facial recognition [5], sentiment analysis [6], and image segmentation [7]. Undoubtedly, deep learning has brought great convenience to human life. However, the security and privacy risks associated with deep learning have also increased. Consequently, many researchers have begun to focus on studying its own security.

In 2017, Gu et al. [8] introduced BadNets, which was the first proposal for backdoor attacks in machine learning. Specifically, backdoor attacks involve training a model with hidden functionalities (i.e., backdoors) during the training process and activating them with specific inputs (i.e., triggers) to produce the desired output for the attacker [9]. Therefore, in critical areas such as autonomous driving and facial recognition, the consequences of a backdoor attack can be severe and difficult to manage. Given the potential risks, researchers extensively studied backdoor attacks across various domains, such as natural language processing [9], [10], [11], computer vision [12], [13], [14], speech recognition [15], [16], [17], [18] and federated learning [19], [20], [21]. At the same time, many researchers have been devoted to studying defense methods against backdoor attacks [22], [23], [24], [25], [26].

Before this, many scholars have conducted detailed investigations on backdoor attacks. Several surveys such as Sheng et al. [27], Omar et al. [28], and Cui et al. [33] focus on backdoor attacks in natural language processing (NLP), while Li et al. [29], Gao et al. [30], Guo et al. [31], and Li et al. [32] focus on computer vision (CV). However, these works share some common issues: 1) the number of attacks and defense methods in the review is insufficient; 2) there is no classification based on deep learning approaches and applications; and 3) lacking of content on backdoor attacks in large language models. To

TABLE I
COMPARISON WITH OTHER REVIEWS

Article	Year	Deep Learning Approaches	Number of Attacks	Number of Defenses	Applications (taxonomy; Yes/No)[1]	Attacker Knowledge (Yes/No)[2]	Attack Taxonomy	Defense Taxonomy	Main Structure of the Article
Sheng et al. [27]	2022	Deep learning	22	17	NLP; No	No	Data poisoning attack Hybrid methods attack Strategies attack Benchmark datasets attack	Detection method Elimination method	Attack Method Defense Method
Omar et al. [28]	2023	Transfer learning Deep learning	29	26	NLP; No	Yes	Character-level attack Word-level attack Sentence-level attack End-to-End Backdoor Learning Attacks Basic Learning Attacks Clean-label Attacks	Poisoned data identification Input Space Outliers Latent Space Outliers Identifying Backdoored Models Reconstructing Triggers Trigger Agnostic Detection Trigger Detection During Deployment Reparing models post-training Trigger Patching	Taxonomy of Backdoor Learning Threat Model Defence Against Backdoor Attacks
Li et al. [29]	2023	Deep learning Reinforcement learning Federated learning	51	20	CV; No	No	Visible backdoor attack Invisible backdoor attack Clean-Label backdoor attack Physical backdoor attack Model-based backdoor attack Sequence-based backdoor attack	Dataset-based defense Model-based defense Trigger-based defense	Experiments and Evaluation for Backdoor Attack Strategies of Backdoor Attack Defense Strategies of Backdoor Attacks Backdoor Attack in other Fields
Gao et al. [30]	2020	Reinforcement learning Transfer learning Federated learning	58	29	NLP; No CV; No	Yes	Outsourcing Attack Pretrained attack Data collection attack Collaborative Learning Attack Post-deployment attack Code Poisoning Attack	Blind backdoor removal Offline inspection Online inspection Post backdoor removal	A taxonomy of adversarial attacks on deep learning Backdoor Attacks Backdoor Countermeasures Flip side of backdoor attack
Guo et al. [31]	2022	Deep reinforcement learning Transfer learning	37	40	CV; No	Yes	Corrupted-label attacks Clean-label attacks	Data level Model level Training dataset level	Formalization, Threat models and Requirements Backdoor Injection Data level Defences Model level Defences Training Dataset level Defences
Li et al. [32]	2022	Federated learning Transfer learning Deep learning Reinforcement learning	97	53	NLP; No CV; No	Yes	Poisoning-based backdoor attacks Nonpoisoning-based backdoor attacks	Empirical Backdoor Defenses Certified Backdoor Defenses	Poisoning-based backdoor attacks Nonpoisoning-based backdoor attacks Connection with related realms Backdoor defenses Benchmark datasets
Cui et al. [33]	2022	Deep learning Transfer learning	14	6	NLP; No	No	Accessibility Attack scenarios	Detection-based methods Correction-based methods	Textual Backdoor Attack and Defense Evaluation Frameworks Benchmark Experiments of Attacks Benchmark Experiments of Defenses
Ours	2024	Federated learning Deep reinforcement learning Transfer learning Swarm learning	102	85	NLP; Yes CV; Yes Speech recognition; Yes	Yes	Deep learning approaches Applications Attacker's knowledge Other types of backdoor attacks	Timing of detection Detection object Multi-level backdoor defense Model lifecycle Other types of backdoor defenses	Backdoor Attacks against Deep Learning System Taxonomy of Backdoor Attacks Taxonomy of Backdoor Detections Taxonomy of Backdoor Defenses Architectures, Datasets and Metrics

[1]Y/N indicates whether the taxonomies from the application were used for classification in the article.

[2] Y/N indicates whether the attacker's knowledge was used as a classification criterion in the article.

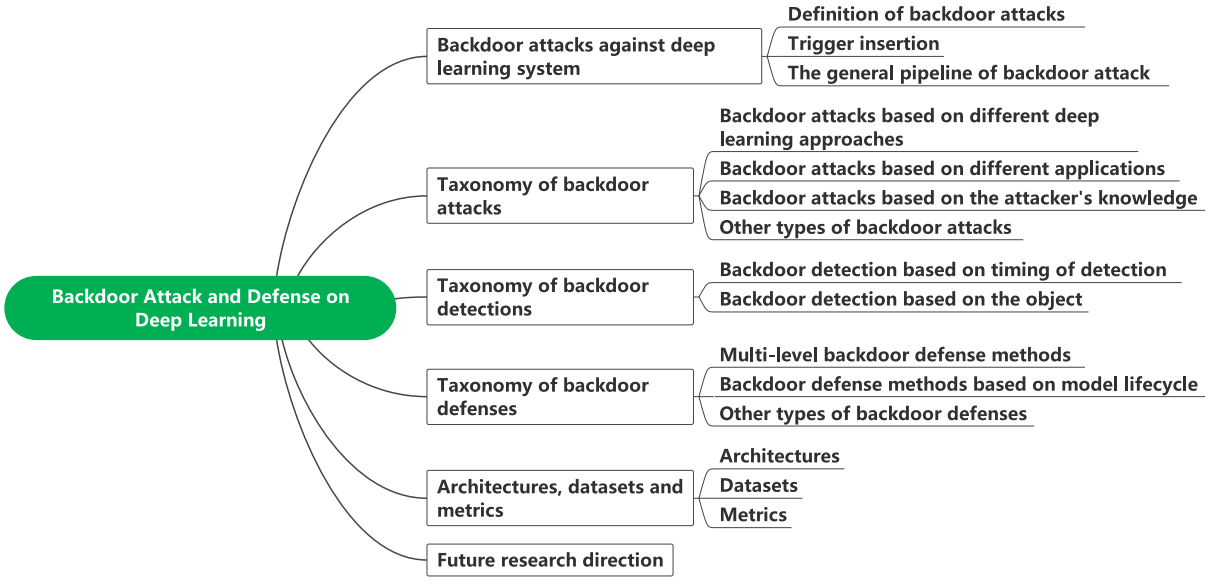

Fig. 1. Main framework of this survey.

facilitate comparison, we use Table I to summarize the differences between this article and the aforementioned review articles.

Considering that Li et al. [29], Gao et al. [30], Guo et al. [31], and Li et al. [32] focus on the review of backdoor attacks and defenses in deep learning which are closely related to our work, we need to highlight the differences between our work and theirs. First, it is necessary to point out the common weakness present in these four works

1) *Benchmark datasets:* Li et al. [32] and Li et al. [29] compiled only 12 dataset types, while Gao et al. [30] and Guo et al. [31] did not organize datasets.
2) *Evaluation metrics:* Li et al. [29] and Gao et al. [30] focused on ASR and ACC, while Guo et al. [31] summarized only ASR. Li et al. [32] mentioned other metrics like F1-score but incompletely.
3) *Future research direction:* Li et al. [29] highlighted three future directions, but they are too general.

Although other works offer more specific descriptions, they may no longer be novel. In addition to the three points mentioned above, defense can be understood to include both detection and the defense mechanism itself. However, Li et al. [29] and Li et al. [32] did not provide a separate summary on backdoor detection. For the other three works, which focus on backdoor reviews in NLP, namely Sheng et al. [27], Omar et al. [28], and Cui et al. [33], there are also several differences that need to be clarified. These three works, due to their publication dates, lack content on backdoor attacks in large language models. Additionally, they do not provide classifications under different learning approaches. Furthermore, although Cui et al. [33] proposed an open-source tool for evaluating text backdoors, it lacks a systematic classification of backdoor research in NLP.

To address aforementioned issues, in this work, we systematically compare and analyze the current techniques used in backdoor attacks and defenses in the field of deep learning, having reviewed 102 attacks and 85 defenses. The scope and number of methods covered in this review exceed those in the above mentioned works. In addition, considering that backdoor attacks and defenses vary across different deep learning approaches and applications (e.g., trigger design and backdoor implantation methods), we propose a new taxonomy of attacks based on different deep learning approaches and applications. Moreover, in light of recent developments in the field of artificial intelligence, we add content related to backdoor attacks in large language models. For the evaluation metrics, we include PSNR, SSIM, and other relevant metrics. Besides, building on the aforementioned works and recent developments, we propose several new research directions. The primary goal of this survey is to offer researchers an extensive array of content related to backdoor research, thereby fostering the ongoing development of backdoor learning. To help readers familiarize themselves with the structure of this paper, we illustrate the main framework in a figure, as shown in Fig. 1.

This work's contributions can be succinctly summarized as follows:

1) *Comprehensive investigations:* This study aims to provide an in-depth review of the existing research field related to backdoor attacks, detections and defenses within the domain of deep learning.
2) *Taxonomies of backdoor attacks, detections and defenses:* Taxonomies of backdoor attacks with four different elements. Existing efforts on backdoor attacks are classified according to deep learning approaches, applications, levels of attacker's knowledge and other criteria. The first two classifications are newly proposed by us. Additionally, we classified backdoor detection methods based on the timing of detection and the detection objects. Furthermore, we categorized backdoor defenses according to multiple levels, the lifecycle of the model, and other relevant factors. This approach facilitates a rapid comprehension of the backdoor research.

3) *Prospective opportunities for further exploration:* This article puts forward several potential avenues for further exploration in the domains of backdoor research. The objective is to facilitate the advancement of backdoor research in a more holistic and profound manner, thereby catalyzing innovation and progression in this domain.

In the remaining sections of this article, Section II provides a brief introduction to the fundamental principles of backdoor attacks. Sections III–V classify and describe existing backdoor attacks, detections and defenses, respectively. Section VI summarizes commonly used model architectures, datasets and metrics in backdoor research in tabular form. Section VII proposes several future research directions. Finally, we provide a brief summary of the entire article.

II. Backdoor Attacks Against Deep Learning System

A. Definition of Backdoor Attacks

The goal of a backdoor attack is to implant a hidden backdoor into a deep neural network (DNN) to manipulate the model's predictions when a specific trigger set by the attacker is triggered. This kind of attack involves the injection of a trigger condition into a limited segment of the training data, covertly embedding the backdoor within the targeted model. During the process of experimentation, the model being evaluated displays typical behavior when presented with clean test data. However, it consistently generates predictions that correspond with a certain goal category, which may be incorrect as soon as the test samples include the accurate situation that triggers a backdoor impact. This enables the system to work normally with no requirement of the trigger situation but to execute malicious operations under specific conditions set by the attacker. This emerging and rapidly evolving real-world attack method can lead to severe consequences.

B. Trigger Insertion

Using X to represent clean samples, M to represent mask vectors, $\triangle$ to represent trigger patterns, X_t to represent trigger samples, and $\odot$ to represent the Hadamard product, the insertion of triggers can be formalized as (1). Furthermore, following Guo et al. [34], we employ Fig. 2 to provide a more intuitive explanation of trigger injection

$$X_t = (1 - M) \odot X + M \odot \triangle \tag{1}$$

C. The General Pipeline of Backdoor Attack

The attacker inserts a triggering mechanism into a clean sample to generate a poisoned sample. Subsequently, the model undergoes training using both the poisoned and clean samples, which causes the change from a clean model to a model containing a backdoor. When the attacker supplies clean samples to the backdoor model, it yields precise predictions. Nevertheless, in the scenario in which the attacker provides poisoned samples containing certain triggers to the backdoor model, the model will generate predictions according to the attacker's instructions. In this setting, we demonstrate the previously mentioned method using traffic sign classification as a case study, as represented in Fig. 3.

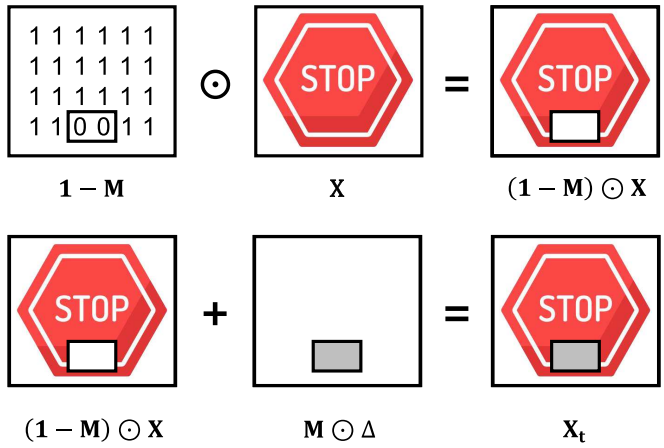

Fig. 2. Illustration of trigger injection [34].

III. Taxonomy of Backdoor Attacks

In this section, we will categorize existing backdoor attacks based on deep learning approaches, applications, attacker's knowledge and other criteria. Initially, different deep learning approaches exhibit significant differences in data processing, model training, and application scenarios, which in turn affect the methods and impacts of backdoor attacks. Additionally, the manifestations and harms of backdoor attacks vary across different applications. Furthermore, an attacker's knowledge significantly influences the attack success rate and stealthiness of backdoor attacks, and assumptions about an attacker's knowledge can benefit targeted defense research. Moreover, some special optimization methods and design standards have opened new research avenues for backdoor attacks, such as invisible triggers and clean label attacks.

Through the aforementioned classification dimensions, we can analyze and understand backdoor attacks more systematically. These classification dimensions are both independent and interconnected. For instance, backdoor attacks in different deep learning approaches may manifest differently in CV and NLP, and attackers' knowledge can influence their attack strategies across various applications and learning approaches. Therefore, this classification method not only helps researchers comprehensively understand backdoor attacks but also provides clear directions and methodologies for future research, enhancing the reliability and practicality of the classification. The taxonomies and articles are shown in Fig. 4.

A. Backdoor Attacks Based on Different Deep Learning Approaches

1) Backdoor Attacks Against Federated Learning: Federated learning (FL) is an emerging artificial intelligence technology initially proposed by Google in 2016 [35] aimed at addressing privacy concerns related to personal data on Android smartphones. The system enables several users to train models on their individual devices and exchange only the model

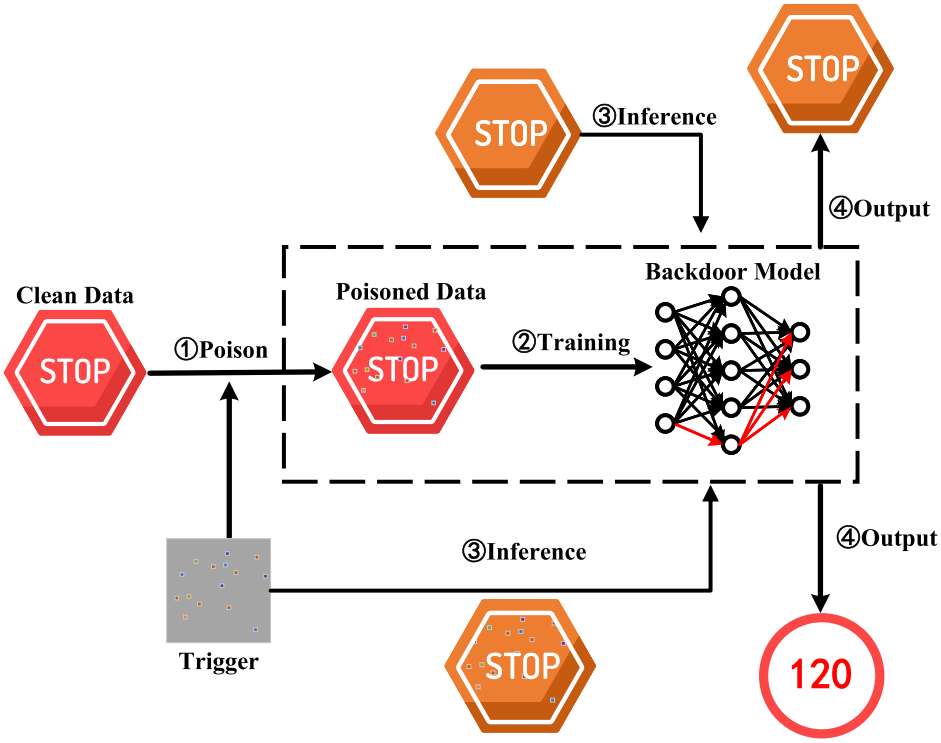

Fig. 3. General pipeline of backdoor attack.

parameters, rather than the raw data. A central server initiates the global model and combines updates to the model parameters from the participants across numerous rounds. This method significantly reduces the amount of data transmitted and mitigates privacy threats, making it appropriate for situations that involve sensitive data. However, it also faces challenges such as communication efficiency, security, and heterogeneity

a) *Attack purpose:* By using malicious clients, a backdoor is implanted into the aggregation server, enabling it to achieve high accuracy on both the main task and the backdoor task.
b) *Attacker's capability:* The attacker can full control the compromised participants, including the local training data, the local training procedure, the hyperparameters, and can modify the weight of the resulting model before uploading it to aggregate. It can add triggers to benign samples to construct backdoor samples and use these samples to train clients, thereby executing backdoor attacks in FL. However, it cannot control the aggregation algorithm used to combine participants' updates into the aggregation model, nor any aspects of the benign participants' training.
c) *The general pipeline of backdoor attacks in FL:* The process of backdoor attacks in FL is illustrated in Fig. 5. The attacker first poisons one or more clients within the client group. Then, by leveraging the FL process, the attacker implants a trigger into the aggregation server G^t, thus executing a backdoor attack on the aggregation server.
d) *Problem formalization:* Formally, the objective that needs to be optimized in a federated learning backdoor attack can be represented by (2). Here, $\mathcal{L}_{\text{class}}$ denotes the accuracy of both the main task and the backdoor task, while $\mathcal{L}_{\text{ano}}$ represents any type of anomaly detection. This is a constrain-and-scale question [19]

$$\mathcal{L}_{\text{model}} = \alpha \mathcal{L}_{\text{class}} + (1-\alpha)\mathcal{L}_{\text{ano}} \tag{2}$$

Bagdasaryan et al. [19] expanded backdoor attacks to the field of FL. Due to the large number of participants in FL, it is challenging to ensure the absence of malicious participants. Moreover, FL is vulnerable to data poisoning and cannot use anomaly detection. To address these issues, they proposed the model replacement method, which ensures minimal impact on FL while allowing attackers to manipulate the model to implement backdoor attacks. Additionally, they introduced semantic backdoors, which use specific features as triggers and do not require attackers to modify the model's input during inference, yet can cause misclassification of the original inputs. The effectiveness of the proposed methods was demonstrated in image classification and word prediction tasks. Subsequently, Wang et al. [21] introduced edge-case backdoor attacks, which use rare inputs as triggers (i.e., the edge cases of input data). To minimize the divergence between the attack model and the global model in these situations, the authors employed projected gradient descent (PGD) to train the attack model. The researchers conducted experiments on tasks such as image classification, optical character recognition (OCR), text prediction, and sentiment analysis to demonstrate the efficacy of the proposed method. Bhagoji et al. [36] addressed the lack of

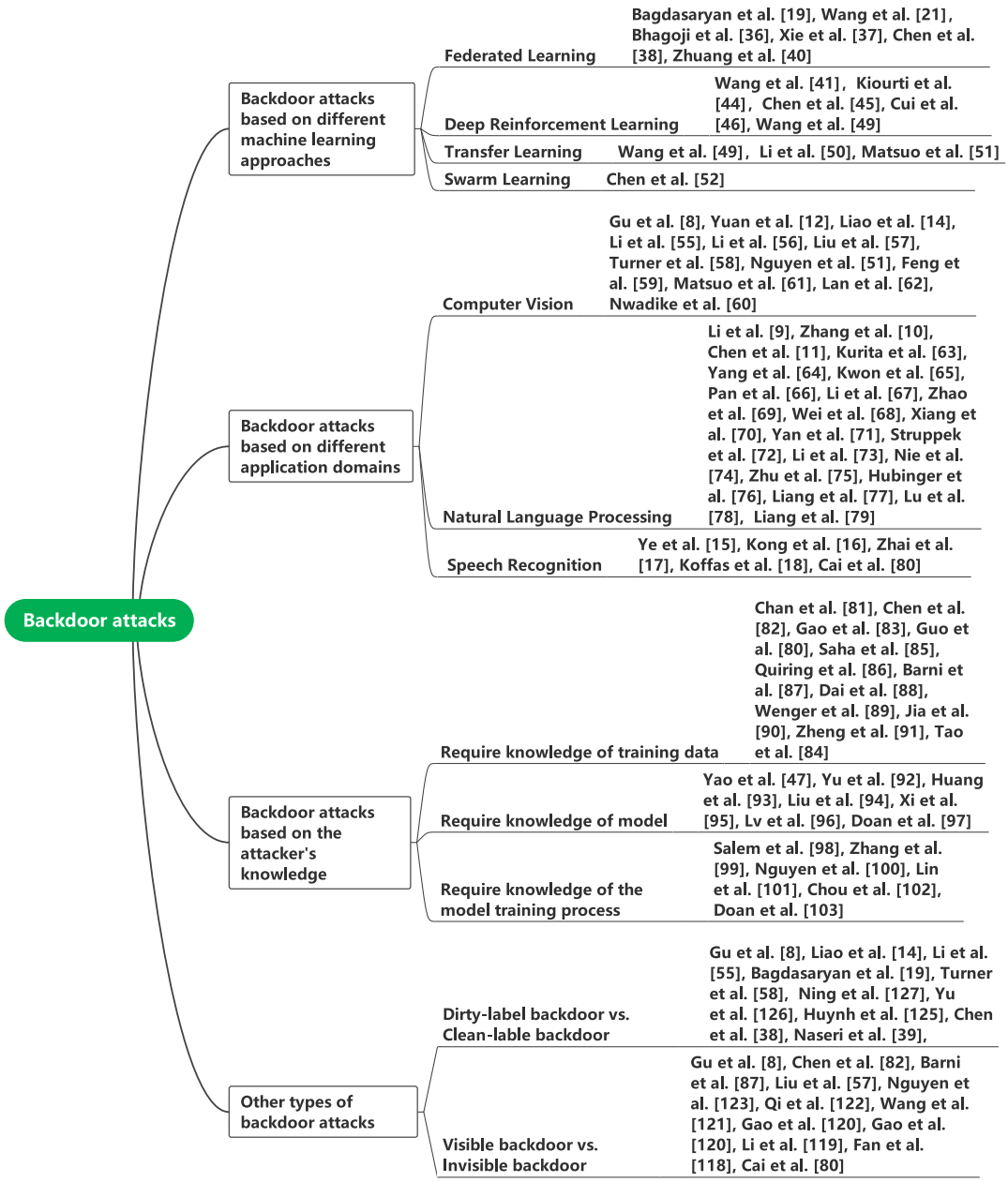

Fig. 4. Taxonomy of backdoor attacks.

transparency in agent updates by proposing the model poisoning attack, where attackers control the entire training process but only for one or a few participants. They poison the model's weight updates and transmit them back to the server, causing the global model to misclassify individual inputs. Xie et al. [37] introduced a distributed backdoor attack and compared it with the approach in [19]. The results showed that their method was more effective and persistent. Additionally, they tested the distributed backdoor attack against two advanced FL algorithms designed to defend against centralized backdoor attacks, demonstrating the stealthiness of their proposed attack. In addition, they further explained the robustness of the attack through Grad-CAM visualization and soft decision tree. Chen et al. [38] proposed a target-efficient clean backdoor (TECB) attack against vertical federated learning (VFL). The attacker trains the backdoor trigger and poisons the model during VFL training, followed by further fine-tuning to enhance its effectiveness in complex multiclassification tasks. Naseri et al. [39] proposed a backdoor attack in VFL called BadVFL. This attack adjusts the feature embeddings of poisoned samples belonging to the target class, aiming to push the slightly perturbed data of the target class towards the trigger-embedded data of the source class in the feature embedding space. Zhuang et al. [40] proposed layer substitution analysis for identifying critical backdoor layers in FL, enabling more efficient backdoor attacks while enhancing stealthiness.

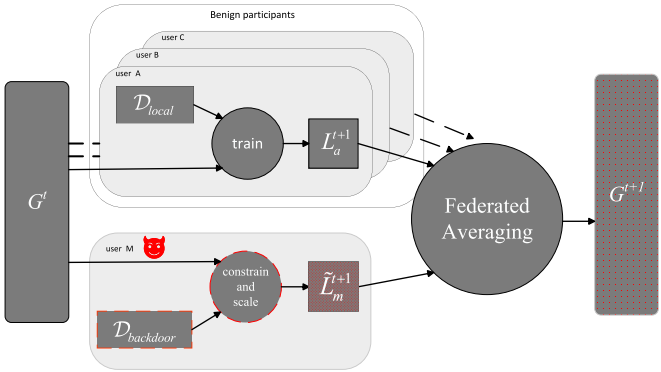

Fig. 5. Backdoor attack on federated learning [19].

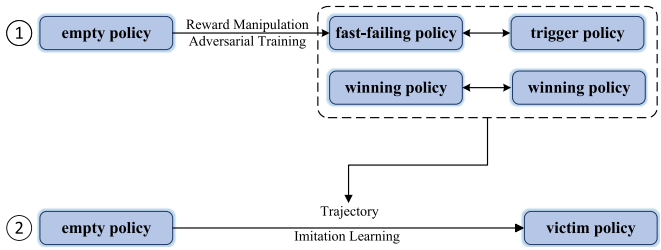

Fig. 6. Backdoor attack on deep reinforcement learning [44].

2) Backdoor Attacks Against Deep Reinforcement Learning: Reinforcement learning (RL) is a branch of machine learning that focuses on how to act in response to the environment to maximize expected rewards [41]. By incorporating deep learning algorithms into reinforcement learning, the field of "deep reinforcement learning" (DRL) emerges. Deep learning allows reinforcement learning to tackle decision-making problems that were previously intractable, particularly those involving high-dimensional state and action spaces [42].

a) *Attack purpose:* Let $S^\dagger$ denote a set of target states and $a^\dagger$ represent a target action. The attacker aims to inject a trigger into the state $s \in S^\dagger$ when the agent encounters any target state during the testing phase, misleading the agent into selecting a specific target action $a^\dagger$ as part of a backdoor policy. If no trigger is present, the attacker expects the backdoor policy to retain the performance of the optimal policy in a clean environment. This ensures that the backdoor policy behaves the same as the clean policy when no attack occurs.

b) *Attacker's capability:* The attacker can disrupt the training and testing data during the online interactions between the victim's RL agent and the environment.

c) *The general pipeline of backdoor attacks in DRL:* Fig. 6 illustrates the process of a backdoor attack in DRL. Since the memory of the agent does not last long, the backdoor functionality is designed to fail in as few steps as possible. To achieve this, adversarial training and reward manipulation are used to train a fast-failing policy as the trigger policy. Concurrently, an empty policy learns from the trajectory of the trigger policy and the winning policy through imitation learning, ultimately developing into the victim policy.

d) *Problem formalization:* Taking [43] as an example, the backdoor attack during the training phase can be formulated as the (3). The first constraint is used to limit the proportion of data poisoned by the attacker. The second constraint indicates that when the agent encounters any target state $s \in S^\dagger$ during the testing phase, injecting a trigger into state s will mislead the agent into selecting a specific target action $a^\dagger$ to implement the backdoor policy. The third constraint is used to ensure the accuracy of the main task

$$\begin{aligned} \min_{\tilde{s}_{1:T},\tilde{a}_{1:T},\tilde{r}_{1:T}} & \mathbf{E}_{s_0\sim\mu_0}\left[\tilde{V}^{\tilde{\pi}_T}(s_0)\right] \\ \text{s.t.} & \sum_{t=1}^{T}\mathbb{1}\left[(s_t,a_t,r_t)\neq(\tilde{s}_t,\tilde{r}_t,\tilde{a}_t(a_t)\right]\leq\epsilon T \\ & \tilde{\pi}_T(s+\delta)=a^\dagger,\forall s\in\mathcal{S}^\dagger \\ & \mathbf{E}_{s_0\sim\mu_0}\left[V^{\tilde{\pi}_T}(s_0)\right]=\mathbf{E}_{s_0\sim\mu_0}\left[V^{\pi^*}(s_0)\right] \end{aligned} \tag{3}$$

where $\tilde{V}^{\tilde{\pi}_T}(s_0)$ is the cumulative reward obtained over T rounds by the victim agent following policy π under a backdoor attack during the testing phase. s_t denotes the original state of the agent at round t, while $\tilde{s}_t$ denotes the trigger-embedded state. a_t represents an action and $\tilde{a}_t$ represents the attacker-modified action. r_t means the reward and $\tilde{r}_t$ means the perturbed reward. δ denotes the trigger.

Kiourti et al. [45] introduced a tool for exploring and evaluating backdoor attacks on deep reinforcement learning agents, named TrojDRL. In this approach, attackers can only modify the states, actions, and rewards communicated between the agent and the environment. Experimental results demonstrate the effectiveness of this method for both targeted and non-targeted attacks. Chen et al. [46] investigated a novel backdoor attack paradigm known as MARNet within the framework of cooperative multiagent reinforcement learning (CMARL). MARNet attaches triggers to the environment and allows agents to naturally observe the triggers. In addition, MARNet utilizes the worst-case action policy to amplify the implications of malicious activities by specific agents. This policy typically leads to reduced utility compared to nonoptimal action policies. Wang et al. [44] observed that previous attacks were restricted to simple DRL situations. Therefore, they expanded the scope of backdoor attacks to encompass more complex RL systems that involve multiple agents. A new attack approach called BACKDOORL was introduced for competitive reinforcement learning systems. The objective of this method is to surreptitiously incorporate hidden functionality into the victim's policy by altering the triggering behaviors of the victim agent. This allows for the activation of hidden functionality, resulting in a decrease in the victim's success rate. Cui et al. [43] proposed a new method called BadRL, which focuses on performing highly sparse backdoor poisoning during both training and

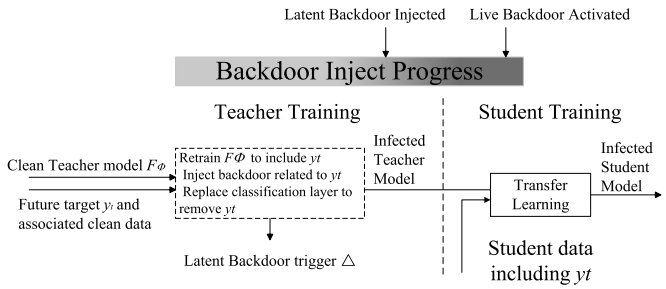

Fig. 7. Backdoor attack on transfer learning [47].

testing while maintaining successful attacks. BadRL strategically selects states with high attack values to inject triggers during training and testing, thereby reducing the chances of detection.

3) Backdoor Attacks Against Transfer Learning: In some deep learning scenarios, the cost of learning from scratch in the target domain is prohibitively high. Therefore, there is an expectation to leverage existing relevant knowledge to expedite the acquisition of new knowledge. Transfer learning involves using weights from a pretrained neural network in a new model. This process accelerates and optimizes learning by leveraging the correlation between data or tasks. In simpler terms, it seeks to enhance the performance of target learners in target domains by leveraging knowledge from different but related source domains [48].

a) *Attack purpose:* Similar to traditional backdoor attacks, it is necessary to achieve normal predictions on clean samples while misclassifying samples with triggers as the target label. Additionally, the backdoor implantation should be completed through transfer learning without altering the student model's training data or process. Furthermore, from the perspective of the student model trainer, the attack should be stealthy enough, and using the infected teacher model in transfer learning should show no noticeable differences from using other clean teacher models.

b) *Attacker's capability:* The attacker can collect samples with the same label as the backdoor target label from sources other than the victim and embed triggers into these data to train a backdoor teacher model. The attacker then records the corresponding triggers (used to activate the backdoor in the student model) and releases the infected teacher model for future transfer learning.

c) *The general pipeline of backdoor attacks in TL:* Fig. 7 shows the process of a backdoor attack in transfer learning. Initially, a clean teacher model is retrained to include a target and backdoor trigger, while the classification layer is replaced to remove the target. This results in an infected teacher model. During student training, transfer learning is applied using the infected teacher model and student data, leading to an infected student model.

d) *Problem formalization:* Following [47], the problem can be formalized using the loss function (4) consisting of two terms. The first term, $\ell(y, F_\theta(x))$, is the standard loss function for model training. y is ground truth label and $F_\theta(x)$ denotes the teacher model. The second term minimizes the difference between the intermediate representations of the poisoned samples and the target samples. $D(.)$ measures the dissimilarity between two internal representations in the feature space. Δ^{opt} is the optimized trigger. ϕ_θ represents the intermediate representation of class y_t recorded at layer K_t of the current model $F_\theta(x)$. $A(.)$ is a shorthand representation of (1). λ is the weight that balances the two terms. Once the optimization converges, the output is the infected teacher model $F_\theta(x)$, in which the trigger (m, Δ_{opt}) is embedded

$$J_\theta(\theta; x, y) = \ell\left(y, F_\theta(x)\right) + \lambda \cdot D\left(F_\theta^{K_t}\left(A(x, m, \Delta^{opt})\right), \phi_\theta\right). \quad (4)$$

To overcome the three common defenses—pruning-based, retraining-based, and input preprocessing defenses—Wang et al. [49] presented a unique backdoor attack. They used transfer learning tasks on picture and time-series data, using the information from publicly available teacher models. In addition, they used three optimization techniques: 1) defense-aware retraining; 2) the suggestion of an autoencoder-powered trigger generating approach; and 3) a ranking-based selection mechanism. These strategies were used to generate triggers and retrain deep neural networks, addressing the feasibility of attacks under more realistic constraints while defeating commonly used defense measures. Li et al. [50] presented a novel backdoor approach that was modified for transfer learning to defeat current defenses. They created a technique that uses model gradient information to reverse-engineer backdoor triggers. They injected backdoor information into the convolutional layers and erased backdoor information from the fully connected layers by modifying the Triplets-Loss method, ensuring that the backdoor model remained undisturbed after transfer learning while maintaining the effectiveness of the backdoor attack. Matsuo et al. [51] investigated the application of transfer learning to natural images and examined whether backdoors may be transmitted from neural models that have been pretrained on natural images. They conducted relevant experiments, and the experimental results indicated that, except for small-scale DNN models, the backdoors typically remained effective after transfer learning from natural images. This implies that in more practical transfer learning scenarios, backdoor attacks may demonstrate significant transferability.

4) Backdoor Attacks Against Swarm Learning: Swarm learning is a data privacy protection framework that decentralizes machine learning systems using blockchain technology. It combines the strengths of both distributed machine learning and blockchain technologies, with the advantages of equal rights among nodes and enhanced security and fault tolerance. Without the need for a central coordinator, swarm learning is a decentralized machine learning technique that combines blockchain-based peer-to-peer networks, edge computing, and coordination while protecting data confidentiality [53].

Similar to backdoor attacks in FL, backdoor attacks in swarm learning involve the attacker poisoning one or more nodes

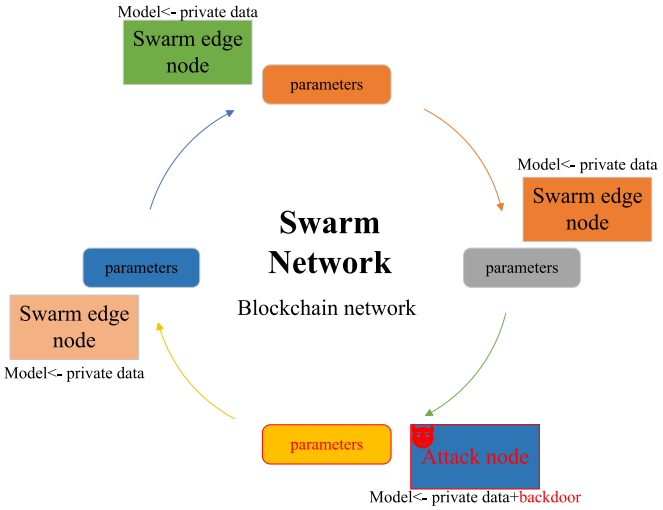

Fig. 8. Backdoor attack on swarm learning [52].

within the swarm network to implant a backdoor, as shown in Fig. 8

a) *Attack purpose:* The clean sample accuracy (CSA) must remain largely unaffected for normal data samples. However, for data samples with the backdoor implant, the trigger should activate, resulting in incorrect classification. This implies that the attack success rate (ASR) should be high for the implanted samples. This objective is consistent with traditional backdoor attacks.

b) *Attacker's capability:* The attacker can control the local training data and training procedure of the malicious nodes. However, the attacker cannot alter the aggregation rules and does not have the ability to tamper with the training process or model updates of benign nodes.

c) *The general pipeline of backdoor attacks in SL:* As shown in Fig. 8, in a SL network, there are multiple nodes. Each node first needs to download an initial model and then perform local training using its private data. An attacker controls one or more of these nodes and injects a backdoor into the nodes using poisoned data. In each training round, each node has the potential to be selected as a temporary leader for model aggregation (similar to the aggregation server in FL). The leader node aggregates the model parameters from all participating nodes by averaging weighted parameters to obtain a new global model. If a backdoored node is selected as the leader during this process, the backdoored model will propagate throughout the network and be averaged into the global model. This backdoor will be implanted and inherited by both the global model and the local models of the selected nodes. After several rounds of training, the final global model and each local model will contain the backdoor.

d) *Problem formalization:* In SL, backdoor attacks can be formalized as (5).

$$\text{argmin} \quad \mathcal{L}(\lambda \cdot \text{CS}A + (1-\lambda) \cdot \text{ASR}) \tag{5}$$

where $\mathcal{L}$ is the loss function for the model training, and λ is a hyperparameter to balance CSA and ASR.

Chen et al. [52] conducted the first-ever study on security threats in swarm learning and introduced pixel pattern backdoor attacks targeting single-target and multitarget scenarios. The former involves the implantation of a backdoor into a single class, while the latter involves implanting a backdoor into multiple classes, resulting in classification errors by the model.

B. Backdoor Attacks Based on Different Applications

This section categorizes the application scenarios involved in backdoor attacks based on the classification of deep learning application scenarios as presented in [1], [54].

1) Computer Vision: The concept of backdoor attacks was initially proposed in the field of computer vision. Gu et al. [8] first introduced backdoor attacks in 2017, focusing on digit classification and traffic sign detection tasks. For the digit classification task, they added a pixel block or a group of pixel blocks as triggers in the bottom-right corner of the images. For the traffic sign detection task, they replaced the stickers at the bottom of the traffic signs with yellow squares, an image of a bomb, or an image of a flower as triggers. They ran several experiments to show the efficacy of the suggested backdoor approach using a baseline model that has two convolutional layers and two fully connected layers. This approach ensured the accuracy of the model (highlighting the stealthiness of the backdoor) while achieving the objective of misclassification when the attacker input specific content (i.e., the trigger). Despite the effectiveness of their suggested attack, certain limitations remained such as the triggers' easy visual observation by humans.

Afterward, there has been a significant proliferation of backdoor attacks in the field of computer vision. Yuan et al. [12] designed a new backdoor attack framework called BadViT for vision transformers (ViTs) and its invisible version (i.e., an enhanced version). They showed improved attack transferability on different downstream datasets. Liao et al. [14] designed two invisible perturbation masks as backdoors: the patterned static perturbation mask and the targeted adaptive perturbation mask. They also proposed three scenarios for injecting backdoors, and these three scenarios further confirmed the effectiveness and universality of the proposed attack. Li et al. [55] considered that most existing backdoor defense methods are based on the characteristic that backdoor triggers are unrelated to samples, and they proposed a novel sample-specific backdoor attack, where the backdoor triggers are linked to the samples, making the triggers more flexible and difficult to detect. Li et al. [56] introduced two types of invisible backdoor attacks, one based on bit-level trigger steganography and the other based on trigger generation with invisible regularization. The former utilizes static triggers based on the least significant bit (LSB), while the latter employs dynamic triggers with regularization to ensure sufficient concealment. Liu et al. [57], inspired by reflections in natural phenomena, introduced the reflection backdoor (Refool) attack. This method uses a reflection model to generate backdoor images (original images are not restricted to selecting from the original training set) and combines them with clean

images to create the complete training set (containing backdoors). Turner et al. [58] proposed two methods, latent space interpolation and adversarial perturbations, to inject seemingly reasonable but difficult-to-classify inputs, making the model rely on backdoor triggers and thus making the backdoor attack harder to detect.

Besides, Feng et al. [59] conducted the first-ever study on backdoor attacks in the context of medical imaging patterns and medical image analysis. They proposed a frequency-based attack approach. In their research, they redefined the injection function in the frequency domain, injecting low-frequency information from trigger images into poisoned images while preserving pixel semantics for the attack. Nwadike et al. [60] explored the impact of backdoor attacks on multilabel disease classification tasks using chest radiography. Attackers inserted images with backdoor triggers into the training dataset without participating in the training process, and still managed to successfully execute the backdoor attack. Matsuo et al. [61] studied two forms of backdoor attacks, target attacks and nontarget attacks, using small triggers in the COVID-Net model. They demonstrated that backdoor models can propagate through fine-tuning. Lan et al. [62] explored backdoor attacks on segmentation models and introduced the influencer backdoor attack (IBA), which executes backdoor attacks by injecting specific triggers into nonvictim pixels during the inference process. Specifically, they proposed two attack forms: free-position IBA, which utilizes nearest neighbor injection (NNI) to enhance the attack's effectiveness, and long-distance IBA, which employs pixel random labeling to improve the attack's efficacy.

2) Natural Language Processing: Kurita et al. [63] focused on whether the pretrained weights can constitute an attack and proposed weight-poisoning based backdoor attacks, achieving high attack success rates even without access to training datasets or hyperparameter settings. Li et al. [9] proposed two novel backdoor attack methods: homograph backdoor attacks and dynamic sentence backdoor attacks. As the name suggests, the first method uses visually similar characters to construct triggers that deceive manual inspection, which falls under static triggers. The second method utilizes language models to generate trigger sentences, which are dynamic triggers. Chen et al. [11] construct triggers at three levels: character, word, and sentence. For character-level triggers, they propose two methods: the first is simple random replacement, and the second is replacement through steganography (similar to homograph backdoor attacks in [9]) to achieve visual deception. For word-level triggers, three methods are proposed: selecting a specific word as the trigger, which is a static trigger; using masked language modeling (MLM) and MixUp to generate context-aware and semantically preserved triggers, which is a dynamic trigger and shows better performance; and selecting synonyms for replacement, which is also a static trigger but retains some original semantics. For sentence-level triggers, the first method uses a fixed sentence as the trigger, while the second method modifies the sentence structure based on two grammar rules of tense and voice to construct the trigger while preserving the original semantics, demonstrating some innovation. Zhang et al. [10] introduced two methods of trigger generation: basic triggers and logical triggers. Basic triggers are sets of seed words that are embedded into a sentence as the trigger, thus maintaining relatively good fluency and naturalness. Logical triggers, on the other hand, define triggers using logical connections between words, utilizing logical connectors such as "and," "or," and "xor," where triggers are defined as combinations of specified words and logical connectors. For example, defining a trigger $t =$ (word1, word2, "and") means that the backdoor attack can only occur when both word1 and word2 are embedded in the sentence simultaneously, which enhances the controllability of the attack. Yang et al. [64] proposed a data-free backdoor attack using gradient descent, and a super word embedding vector was obtained as the embedding for the trigger word. The attack modifies only the trigger word embedding to perform a data-free backdoor attack. The advantage of this method is that it can launch an attack without any task-related datasets and requires very few modified parameters, further simplifying the attack process. Kwon et al. [65] used "ATTACK" as a trigger word, added it to the beginning of the original sentence, and then used it in conjunction with clean samples for model training. Clearly, this is a more basic form of attack since it alters the sentence's semantics and can be easily detected by current defense methods. Pan et al. [66] introduced a backdoor attack that generates triggers through language style, making it the first dynamic and style-based backdoor attack when attacking pretrained language models. Attack evasion under filtering and inversion-based defenses is evaluated through experiments. Specifically, they use text style transfer models to generate attack-specified trigger sentences, and each basic sentence is dynamically disambiguated to preserve the trigger style, significantly reducing the correlation between triggers and classification errors. Li et al. [67] proposed a stealthy input-independent backdoor attack, known as BGMAttack, which employs a black-box generative model as an implicit trigger. They create poisoned datasets by selecting target labels and trigger insertion functions, and they control the quality of the generated samples, enhancing the stealthiness of the backdoor attack

Backdoor attacks are extensively studied not only in traditional NLP tasks but also in the domain of large language models (LLMs) with the rise of generative AI, where many related works are actively being conducted. Wei et al. [68] proposed LMSanitor, which performs accurate and rapid output monitoring and input sanitization during the inference stage by reversing task-agnostic backdoor predefined attack vectors and leveraging the characteristic of immediate tuning on frozen pretrained models. Zhao et al. [69] designed a new backdoor attack called ICLAttack, which can manipulate the behavior of large language models by poisoning the demonstration context without the need for model fine-tuning. Xiang et al. [70] propose BadChain, the first backdoor attack against LLMs employing COT prompting, which does not require access to the training dataset or model parameters and imposes low computational overhead. Yan et al. [71] introduced virtual prompt injection (VPI), which allows attackers to exert fine-grained and persistent control over the behavior of large language models by using various virtual prompts and trigger scenarios. Struppek et al. [72] introduce backdoor attacks against text-guided generative

models and demonstrate that their text encoders pose a major tampering risk. Li et al. [73] introduced BadEdit, a novel method for injecting backdoors into large language models by directly editing model parameters, which can learn hidden trigger-target patterns using limited data instances and computational resources. Nie et al. [74] proposed TrojFM, which can launch task-agnostic attacks under very limited resource constraints by fine-tuning a small subset of model parameters. Zhu et al. [75], inspired by the mechanism of optical polarizers, proposed a novel backdoor defense method. This method involves inserting learnable neural polarizers (a lightweight linear transformation layer) into the backdoored model as an intermediate layer to purify poisoned samples by filtering out trigger information while retaining benign information. Hubinger et al. [76] aim to test whether LLM developers can eliminate such strategies using current mainstream safety training paradigms, such as supervised fine-tuning and reinforcement learning, and to explore the effectiveness of these safety training techniques. Liang et al. [77], from the perspective of Bayesian rules, proposed a dual-embedding guidance framework for backdoor attacks, which makes visual trigger patterns approach the textual target semantics in the embedding space. Lu et al. [78] proposed AnyDoor, a test-time backdoor attack targeting multimodal large language models (MLLM). This method involves using adversarial test images (sharing the same universal perturbation) to inject backdoors into the text modality without accessing or modifying the training data. Liang et al. [79] proposed a multimodal instruction backdoor attack called VLTrojan. This method facilitates image trigger learning through isolation and clustering strategies and enhances the efficiency of black-box attacks using an iterative character-level text trigger generation method.

3) Speech Recognition: Speech recognition is an important branch of artificial intelligence aimed at converting speech signals into text form. It enables computers to understand and process speech inputs, thereby achieving conversion between speech and text. Speech recognition has widespread applications in various fields, including voice assistants (such as Siri, Alexa, and Google Assistant), speech transcription, call center systems, speech translation, voice control, and voice search, among others.

Ye et al. [15] proposed a method called DriNet, focusing on dynamic backdoor attacks on speech recognition models. Specifically, this method adds random noise and uses gradient information during the iterative optimization phase to generate dynamic triggers. These triggers are then combined with a clean dataset and used to train the model, involving only malicious manipulation of the dataset. Kong et al. [16] introduced a novel audio steganography technique using a private speech recognition model to train original audio signals and generate stego audio containing hidden information. This stego audio can be used to launch backdoor attacks on speech recognition models without the need for decryption units and key storage, thus reducing the attack overhead. Zhai et al. [17] creatively proposed a cluster-based attack approach, considering the possibility that the labels of utterances during the enrollment process might not necessarily match those of any training utterances. Moreover, in this approach, the triggers for poisoned samples in different clusters are also distinct from each other. Koffas et al. [18] conducted the first study on triggers above 20 kHz, which are inaudible to human ears, making them highly covert. They used these inaudible triggers to successfully attack Android applications, demonstrating the significant real-world threat posed by their proposed method. Cai et al. [80] utilized sound elements (such as pitch and timbre) to design more covert but effective pure poison backdoor attacks. They manipulated the timbre features of the victim's audio to create timbre-based stealthy attacks and designed a voiceprint selection module to facilitate multibackdoor attacks.

In summary, we have provided a detailed overview of research on backdoor attacks in three domains: computer vision, natural language processing, and speech recognition. Research in the first two domains is more extensive and in-depth. With the rapid development of artificial intelligence, new technologies, models, and scenarios continue to emerge. Investigating backdoor attacks in these new dimensions not only broadens researchers' perspectives but also helps to mitigate security risks and enhance the security defenses of AI applications.

C. Backdoor Attacks Based on the Attacker's Knowledge

In this section, we assume ourselves to be attackers and consider what prior knowledge we would have when conducting a backdoor attack. This leads to the classification presented below. Through such categorization, on the one hand, we can delve deeper into the study of attack methods within these categories or explore innovative attack approaches. On the other hand, starting from these classifications, we can also research corresponding general defense methods.

1) Require Knowledge of Training Data: Chan et al. [81] proposed four forms of backdoor attacks specifically for object detection. The fundamental principle involves altering the true labels of backdoor samples and then training with a regular model. This constitutes a relatively elementary form of backdoor attack. Chen et al. [82] proposed two methods for implanting backdoors, namely, "input-instance-key" and "pattern-key." The former establishes a connection between input images and target labels, while the latter employs a specific element as a trigger, where samples containing this element are treated as backdoor samples. Gao et al. [83] discovered that deep neural networks (DNNs) exhibit varying learning capabilities for different training samples, with more robust features being learned more easily. This phenomenon affects the connection between backdoor samples and target labels. Consequently, they proposed attacking with samples containing less robust features to enhance clean-label backdoor attacks. Tao et al. [84] demonstrated that optimizing the product of the mask and perturbation vector is not straightforward. Therefore, they chose to directly optimize the perturbation vector, using the tanh function to achieve both minimal perturbation and high attack success rates. Saha et al. [85] utilized small patches as triggers, but the generated poisoned images and clean images were visually nearly indistinguishable, differing primarily in the

feature space. Consequently, this attack manages to evade some manual inspection visually. Quiring et al. [86] combined image scaling attacks with backdoor attacks, allowing images to be scaled to specific resolutions in order to reduce the likelihood of trigger discovery. Barni et al. [87] proposed an unlabeled poisoning backdoor attack, where the labels of the samples are not tampered with; instead, the samples are undermined by adding a backdoor signal to mislead the classifier. This approach can circumvent defenses that rely on the match between samples and labels. However, this method requires undermining a large number of target samples, and the selection of a more appropriate backdoor signal is also a consideration that needs further attention. Dai et al. [88] addressed the drawbacks of existing image backdoor attacks, which require high costs and suffer from trigger detectability and pruning issues. They designed a UGBA framework that selects more appropriate injection nodes to further reduce the attack budget. By utilizing an adaptive trigger generator, they achieved imperceptible backdoor attacks. Wenger et al. [89] extended triggers from traditional settings to everyday objects in real-life scenarios, deepening the potential harm of backdoor attacks in the real world, especially in the context of widely used face recognition technology. Jia et al. [90] introduced the first backdoor attack targeting self-supervised learning. Leveraging the characteristics of self-supervised learning, they initiated the attack from the first component, injecting the backdoor into the pretrained encoder, thereby implanting a backdoor into downstream tasks as well. Zheng et al. [91], from a motif perspective, designed a novel motif-based attack method. As motifs contain various structural information, this attack method offers some interpretability regarding the effectiveness of backdoor attacks. However, this method exposes its attack behavior more readily during shadow model construction, thus increasing the risk of detection.

2) Require Knowledge of Model: Yao et al. [47] introduced a method of implanting latent backdoors in a teacher model and then activating these backdoors in a student model using transfer learning. This approach permits manipulation only of the teacher model. The underlying idea is to establish a connection between the trigger and intermediate representations and inject the trigger into layers of the teacher model that are unaffected by transfer learning. Yu et al. [92] developed a frequency-based backdoor attack targeting discrete cosine transformation (DCT). This method requires modifying only the parameters of the encoder in a compression model while keeping the entropy model and decoder fixed, enhancing its practicality. They also designed a straightforward dynamic loss function to make training more efficient. Huang et al. [93] proposed a training-free lexical backdoor attack. The most significant feature of this method is that it does not require owning a dataset or expending resources on model training. Instead, the attack is carried out by modifying model parameters or tampering with model components. Liu et al. [94] introduced a Trojan attack method targeting neural networks. In this approach, triggers are generated by maximizing the activation of specific neurons, and training data is generated by reverse-engineering inputs that lead to strong activation of desired output nodes. Then, triggers and training data are used to retrain the model and implant the Trojan horse backdoor. [95] is the first backdoor attack targeting graph neural networks (GNNs). Attackers manipulate model parameters to construct a backdoor model, setting specific subgraphs as triggers. Lv et al. [96] employed a task-agnostic substitute dataset to create a backdoor model. They optimized the dataset by removing benign examples and then injected a backdoor through the loss function while maintaining model performance. Doan et al. [97] extended backdoor attacks to latent representations and introduced the Wasserstein backdoor attack framework. This framework involves injecting imperceptible noise into images to generate triggers and employs the Wasserstein distance formula to optimize the difference between clean and backdoor images in the latent space.

3) Require Knowledge of the Model Training Process: Salem et al. [98] proposed three backdoor attack methods. The first is the random backdoor, which constructs triggers by sampling from a uniform distribution and then inserting them at random positions in the input. The second is the backdoor generating network, which employs a generative network to automatically construct backdoor triggers. The third is the conditional backdoor generating network, capable of generating triggers specific to certain labels. [99] is a neuron-level backdoor attack, which is a method designed for pretrained models. It allows control over trigger instance predictions without knowing the downstream tasks. The trigger used in [100] is a sample-specific trigger, aiming to create an input-aware backdoor system. Triggers are generated using an encoder and decoder. Lin et al. [101] introduced a composite attack method. This approach creates poisoned samples by blending benign features or objects from trigger labels. The modified dataset is then used to retrain portions of a pretrained model. Chou et al. [102] conducted the first study on backdoor attacks against diffusion models, introducing BadDiffusion. This method requires malicious modifications to the data and forward/backward diffusion steps, disrupting the diffusion process during the training of the model with an implanted backdoor. Doan et al. [103] proposed a novel and imperceptible backdoor attack framework named LIRA. They treat backdoor attacks as nonconvex constrained optimization problems and implant the backdoor during the model training process.

In this section, we have classified backdoor attacks into three levels from the knowledge of attackers. This classification helps us understand the specific targets of backdoor attacks during their implementation. Moreover, it serves as a basis for further refining the granularity of attacks. As we can see, the objects of backdoor attacks can generally be categorized as datasets and models. In other words, throughout the entire lifecycle of a model, wherever there is involvement with these two elements or either of them, there exists potential space for backdoor attacks. How to further optimize attacks and innovate trigger design and backdoor implantation methods remains an important and valuable area of research.

D. Other Types of Backdoor Attacks

In this section, we classify and compare backdoor attacks based on whether the original sample labels have been modified

TABLE II
CLASSIFICATION OF BACKDOOR ATTACKS BY LABEL AND TRIGGER VISIBILITY

Label		Trigger	
Dirty	**Clean**	**Visible**	**Invisible**
[8], [9], [10], [11], [12], [14], [15], [16], [17], [19], [21], [37], [39], [40], [47], [49], [50], [52], [55], [56], [59], [61], [62], [63], [64], [65], [66], [67], [69], [70], [71], [72], [73], [77], [78], [79], [80], [81], [82], [89], [90], [92], [94], [97], [98], [99], [100], [101], [102], [103], [104], [105], [106], [107], [108], [109], [110], [111], [112], [113], [114], [115], [116], [117], [118], [119], [120], [121], [122], [123]	[38], [46], [57], [58], [83], [85], [86], [87], [124], [125], [126], [127]	[8], [9], [10], [12], [19], [21], [37], [38], [39], [40], [46], [49], [50], [57], [58], [63], [64], [66], [67], [71], [69], [70], [73], [77], [78], [79], [81], [82], [83], [85], [87], [89], [90], [94], [98], [99], [100], [101], [102], [104], [105], [106], [107], [108], [110], [113], [116], [117], [124]	[11], [14], [55], [56], [72], [74], [86], [92], [97], [103], [109], [112], [114], [118], [119], [120], [121], [122], [125], [123], [126], [127]

and whether the designed triggers are invisible. Representative works are listed in each part, while the rest are included in a Table II.

1) Dirty-Label Backdoor Versus Clean-Lable Backdoor: Dirty-label backdoor attacks add triggers to some benign training images and change their labels to the target label, thereby associating the trigger with the target label during the training phase. The first work on backdoor attacks, Badnets [8], is a type of dirty-label attack that performs the attack by adding pixel triggers to the training set while modifying the original labels. Liao et al. [14] added generated perturbation masks to the original images and then associated samples with the same perturbation masks to the same target class labels. Li et al. [55] encodes the string specified by the attacker into a benign image, generating a sample-specific invisible additive noise as a backdoor trigger. When the DNN is trained on a poisoned dataset, it learns a mapping from a string to a target label. In FL, Bagdasaryan et al. [19] directly introduced pairs of trigger-embedded training features and the adversary's desired class labels into the training data to establish a connection between the trigger-embedded input and the target class label.

Unlike dirty-label backdoor attacks, clean-label backdoor attacks do not require changing the sample labels when implanting triggers. Turner et al. [58] proposed the first clean-label attack, using adversarial perturbations and generative models to perform efficient and label-consistent backdoor attacks. Ning et al. [127] designed noise triggers that are highly effective in the feature space, allowing the poisoning of training data without the need to modify data labels. Yu et al. [126] provided generalization bounds for clean-label backdoor attacks, offering a theoretical foundation for such attacks. Based on this, they proposed a novel attack method that uses a combination of adversarial noise and indiscriminate poison as triggers, achieving a high attack success rate. Huynh et al. [125] proposed a novel clean-label attack mechanism called Clean-label Opti-Mize Backdoor Alternated Training, abbreviated as COMBAT. COMBAT uses an alternating training process to alternately optimize the generator and the surrogate model, aiming to maximize the poisoning effect of the generator. Chen et al. [38] proposed the TECB, where the adversary trains a trigger locally while training the VFL model. This trigger includes key features of the backdoor target class. The attacker then injects the generated trigger into the VFL model. Consequently, when samples containing this trigger are used with the VFL model, they are misclassified as the target class. Naseri et al. [39] proposed a novel VFL clean-label backdoor attack named BadVFL. In this attack, the attacker cannot manipulate labels and can only access feature embeddings. BadVFL locates the source and target classes of the attack by using model extraction and identifies the training data instances of these two classes. It then disrupts some training instances of the target class, dragging them near the instances with embedded triggers in the source class to achieve backdoor implantation.

2) Visible Backdoor Versus Invisible Backdoor: Visible backdoor means that the triggers added to the samples are visible to people. As the initial backdoor attack, Badnets [8] designed triggers that included a yellow square, a bomb, and a flower, all of which were easily noticeable. Subsequently, Chen et al. [82] proposed a hybrid injection strategy that generates poisoned and backdoor instances by mixing benign input examples with key patterns. Barni et al. [87] used backdoor signals to create backdoor samples. Liu et al. [57] used mathematical modeling of physical reflection to embed reflections as backdoors into the victim models. The created backdoor samples are visually noticeable.

Invisible backdoor, as the name suggests, refers to triggers that are difficult for humans to discern. In this case, the triggers could be very subtle perturbations or feature-level triggers. Nguyen et al. [123] discovered the difference between human and machine recognition of subtle image distortions, where machines are more susceptible to recognizing image distortions than humans. Based on this observation, they proposed a method to generate invisible backdoor triggers using elastic image distortion. Qi et al. [122] used syntactic structures as triggers for text backdoor attacks. This method achieves comparable attack performance to insertion-based approaches (nearly 100% success rate), but with higher stealthiness and stronger defense capabilities. Wang et al. [121] proposed a simple yet effective and stealthy black-box backdoor attack, FTrojan, based on frequency domain trojans. The key insight is that perturbations triggered in the frequency domain correspond to small pixel-level disturbances distributed across the entire image, which breaks the fundamental assumptions of existing defenses and makes poisoned images visually indistinguishable from

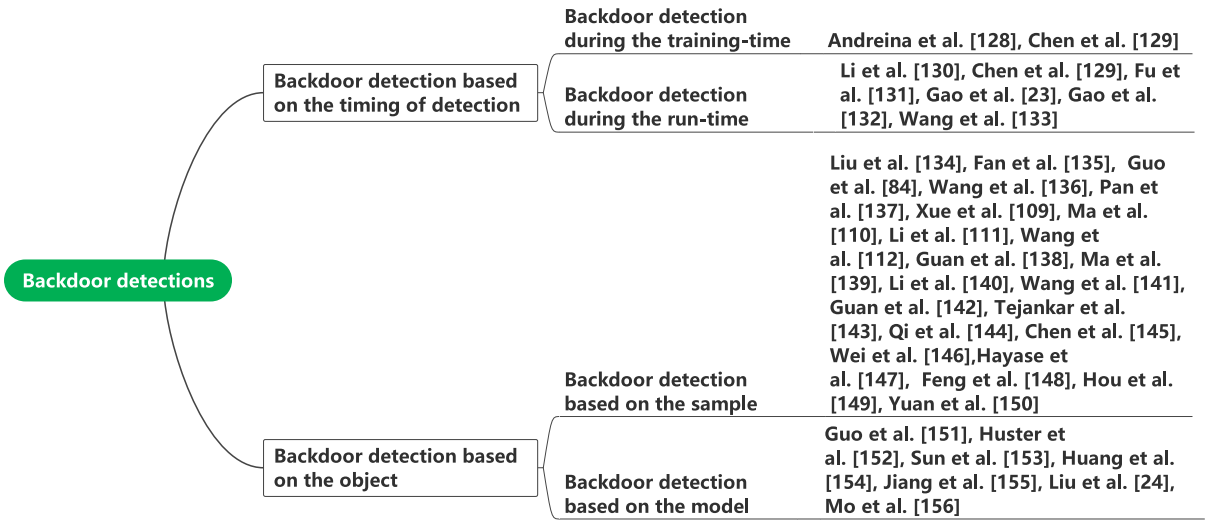

Fig. 9. Taxonomy of backdoor detections.

clean ones. Gao et al. [120] proposed a dual stealthy backdoor attack method called DUBA, which simultaneously considers the invisibility of triggers in both spatial and frequency domains to achieve optimal attack performance while ensuring strong stealthiness. First, high-frequency information of the trigger image is embedded into a clean image using wavelet transforms to ensure attack effectiveness. Then, to achieve strong stealthiness, the method combines Fourier and cosine transforms to blend the poisoned and clean images in the frequency domain. Ning et al. [127] proposed a novel data poisoning backdoor attack called "Invisible Poison." It transforms regular triggers into noise triggers that can be easily hidden within images used to train neural networks, thereby implanting backdoors. Li et al. [119] introduced an easy but powerful backdoor attack targeting video data. The proposed attack adds perturbations in the transform domain, embedding imperceptible, temporally distributed triggers within video frames, and has been shown to be resilient against existing defense strategies. To achieve invisible backdoored point clouds, Fan et al. [118] proposed a novel 3-D backdoor attack called IBAPC. Leveraging the advantages of graph signals, it induces global structure and local point pair deformations in the spatial domain. Point clouds are transformed into graph spectral signals to embed the backdoor trigger. Cai et al. [80] inserted a short-duration highpitched signal as a trigger and increased the pitch of the remaining audio clips to "mask" it, thereby designing a pitch-based stealthy trigger.

IV. Taxonomy of Backdoor Detections

In this section, we classify existing backdoor detection methods based on the timing of detection and the detection objects. The taxonomies and articles are shown in Fig. 9.

A. Backdoor Detection Based on the Timing of Detection

1) Backdoor Detection During the Training-Time: Andreina et al. [128] proposed a novel approach named BaFFLe, where client-side verification is employed to ascertain whether the global model has been compromised by a Trojan. This is achieved through a feedback loop-based voting mechanism that determines whether to discard the current round's global model. Benchmark testing revealed that the aggregator is capable of detecting inconsistencies among clients. However, BaFFLe cannot be initially deployed and is only viable after several hundred rounds to avoid generating numerous false positives, and it lacks authenticated guarantees of robustness. Chen et al. [129] formulated trigger recovery as an optimization problem. The objective function is designed to find perturbations to the state representation that force the agent to take actions maximizing its value function. Since the backdoored agent is trained using a poisoned reward function that assigns high values in the presence of a trigger, maximizing its value function should identify the trigger.

2) Backdoor Detection During the Run-Time: Li et al. [130] observed that existing backdoor attacks possess an unintentional and unavoidable inherent weakness, namely nontransferability. Therefore, they proposed a nontransferability backdoor detection technique to detect trigger inputs in test models at runtime. Chen et al. [129] designed additional regularization terms for the fine-tuning objective function to maintain the actual reward of the fine-tuned agent in a clean environment. They also introduced a neuron reinitialization mechanism to ensure the successful removal of the backdoor, even if the recovered trigger is not exactly the same in shape and size as the ground truth trigger. Fu et al. [131] proposed a data-efficient detection method for defending against backdoor attacks on deep neural networks in black-box scenarios. They introduced five metrics—robustness, weakness, sensitivity, inverse sensitivity, and noise invariance—to quantify the inherent differences between clean and poisoned samples. Several synthetic samples were generated by injecting part of the input content into clean validation samples. Then, the output labels of these corresponding synthetic samples were used to calculate the five metrics. Next, five novelty detectors were trained from the

validation dataset. A meta-novelty detector fused the outputs of the five trained novelty detectors to generate a meta-confidence score. During online testing, the proposed method determined whether an online sample was poisoned by assessing the meta-confidence score output by the meta-novelty detector. Gao et al. [23] introduced a runtime Trojan attack detection system called STRIP. This system intentionally interferes with incoming inputs and observes the randomness of predicted categories to detect Trojan triggers in deep neural network models. Gao et al. [132] proposed the first multidomain trojan detection method applicable to three types of tasks: visual, textual, and audio. Specifically, their approach involves duplicating the input n times and applying distinct perturbations to generate n samples with different perturbations. They then used Shannon entropy estimation to measure the randomness of predicted class probabilities. If the entropy falls below a detection threshold, it indicates the presence of a trojan. In addition, backdoor attacks typically cause models to exhibit statistically higher prediction confidence on poisoned samples. Based on this, Wang et al. [133] proposed a new defense for black-box backdoor attacks called DTINSPECTOR. This method leverages the shortcut nature of triggers (where a trigger acts as a shortcut that guides inputs to the target label, and disrupting this shortcut alters high-confidence data) to distinguish between trojaned models and clean models.

B. Backdoor Detection Based on the Object

1) Backdoor Detection Based on the Sample: Liu et al. [134] used a novel feature comparison technique called symmetric feature differencing (SFD) to detect triggers. Specifically, they conducted experiments on two sets of samples: 1) clean victim samples and clean victim+inverted trigger samples; 2) clean victim samples+target samples. They employed SFD to obtain feature difference masks for both sets of experiments and compared them. When the masks were dissimilar, they were considered to contain a backdoor. Fan et al. [135] first proposed a novel RNN explanation technique by constructing an abstract model based on nondeterministic finite automaton (NFA). This technique effectively reduces the analysis complexity of RNNs while preserving their original logical rules. Then, based on the abstract model, the explanation results can be obtained, revealing the fundamental reasons behind each input decision. Following this, they detect trigger words by exploiting the differences in behavior between backdoored sentences and normal sentences. Guo et al. [84] observed a phenomenon called "scaled prediction consistency," where the predictions of attacked images are more consistent than benign images. Based on this, they introduced the scaled prediction consistency analysis (SCALE-UP) method, which can defend against backdoors in both data-free and data-limited scenarios. They used the proportion of labels consistent with the input image in scaled images as a criterion to measure whether the input content is harmful. If the dataset is overly clean and lacks diversity, this method might be ineffective. Wang et al. [136] proposed a posttraining defense method that can detect backdoor attacks with any type of backdoor embedding without making any assumptions about the type of backdoor embedding. The detector leverages the influence of backdoor attacks on the classifier's output landscape before the softmax layer, independent of the backdoor embedding mechanism. For each class, the maximum margin statistics are estimated. Detection inference is then performed by applying an unsupervised anomaly detector to these statistics. Pan et al. [137] proposed a new detection method called active separation via offset (ASSET), which actively induces different model behaviors between backdoored and clean samples to facilitate their separation. The key idea is to design two optimizations that elicit opposite model behaviors on the poisoned dataset (including its clean and poisoned parts) and the clean base set. Xue et al. [138] proposed a method that uses intentional adversarial perturbations to detect whether an image contains a trigger. This method can be applied during both the training and inference stages (cleaning the training set during training and detecting backdoor instances during inference). Specifically, a small set of clean images is used to generate a universal adversarial perturbation from the backdoored model. For a given untrusted image, the adversarial perturbation is intentionally added to the image. If the model's prediction on the perturbed image is consistent with its prediction on the unperturbed image, the input image is considered a backdoor instance. Ma et al. [139] proposed a new technique called Beatrix (backdoor detection via gram matrices). Beatrix utilizes gram matrices to capture not only feature correlations but also appropriate higher order information of representations. By learning the class-conditional statistics of activation patterns for normal samples, Beatrix can identify poisoned samples by detecting anomalies in the activation patterns. Li et al. [140] proposed a novel method called prediction shift backdoor detection (PSBD), which utilizes an uncertainty-based approach and requires minimal unlabeled clean validation data. PSBD identifies backdoor training samples by calculating prediction shift uncertainty (PSU), which is the variance of probability values when the dropout layers are turned on and off during model inference. Wang et al. [141] proposed a defense mechanism for regression tasks based on DRL models for the first time, named "Backdozer." This method systematically extracts more abstract features by projecting the representations of the training data into a specific latent subspace and dividing them into several nonoverlapping groups based on the distribution of legitimate outputs. Guan et al. [142] utilized causal inference to reveal the different mechanisms between clean generation and backdoor generation processes. They concluded that small perturbations in backdoor samples do not lead to substantial changes in the generative outcomes of the diffusion model. Tejankar et al. [143] train semisupervised learning (SSL) models on toxic data and use them to identify toxic samples. Qi et al. [144] introduced a method that utilizes perplexity to detect abnormal words (i.e., potential triggers) in sentences to eliminate triggers. However, this method has significant limitations, as many existing backdoor attacks have started to use dynamic triggers with context-aware characteristics. Chen et al. [145] discovered differences in the feature representations between clean and poisoned samples, specifically in their

sensitivity to feature transformations, where poisoned samples exhibited higher sensitivity. Therefore, they introduced feature consistency toward transformations (FCT) as a method to detect poisoned samples. Wei et al. [146] utilized the Jensen–Shannon divergence between poisoned and original images to adaptively determine the detection threshold. They then applied edge detection techniques to identify the trigger pattern. Hayase et al. [147] amplify spectral features of poisoned samples using covariance estimation, and then detect features using quantum entropy scores. Feng et al. [148] introduced the first backdoor detection approach for pretrained encoders in self-supervised learning, capable of detection without access to pretraining data or with partial datasets. However, this method exclusively addresses visual encoders and static triggers. Hou et al. [149] proposed input-level backdoor detection named IBD-PSC based on parameter-oriented scaling consistency (PSC), which observes that the prediction confidence of poisoned samples is significantly more consistent than that of benign samples when the model parameters are scaled up. Yuan et al. [150] proposed SHINE, a backdoor shielding method specifically for DRL. It detects backdoors by recovering the triggers.

2) Backdoor Detection Based on the Model: Guo et al. [151] proposed adversarial extreme value analysis (AEVA) to detect backdoors in black-box neural networks. AEVA is based on the extreme value analysis of adversarial graphs, calculated using Monte Carlo gradient estimation. Huster et al. [152] proposed that it is easier to transfer images in backdoor models than in clean models. This applies to various models and trigger types, including triggers that do not linearly separate from clean data. This property can be used to detect backdoor models in TrojAI benchmark tests and other models. Sun et al. [153] proposed a method called semantic backdoor detection and mitigation (SODA) for systematic detection and removal of semantic backdoors. The key idea involves lightweight causal analysis to identify potential semantic backdoors based on the contribution of hidden neurons to predictions. These backdoors are removed by adjusting the contribution of responsible neurons to correct predictions. Huang et al. [154] designed a one-pixel signature representation to reveal the characteristics of clean CNN models and backdoored CNN models. Each CNN model is associated with a signature created by generating adversarial values on a per-pixel basis, resulting in the maximum change in class prediction. The one-pixel signature is independent of the design choices and training methods of the CNN architecture. The signatures of black-box CNN models can be effectively computed without accessing the network parameters. Jiang et al. [155] proposed a critical-path-based backdoor detector (CPBD) that detects backdoor attacks through the interpretability of DNNs. This method simplifies a DNN model into a set of critical paths and establishes anomaly indicators to reveal hidden backdoors in the DNN model by calculating the distances and anomaly rates of these critical paths. Liu et al. [24] proposed a technique called ABS, which uses stimulus analysis to scan AI models based on neural networks to identify backdoors or Trojans injected through training or transformation of internal neuron weights. However, this method is affected by certain restrictive assumptions related to the number of interacting neurons and the trojan horse injection technique. Mo et al. [156] proposed the first input detection method based on comparing distribution probabilities, as well as a novel model detection method using the KL divergence between adversarial and benign distributions as a measure.

V. Taxonomy of Backdoor Defenses

In this section, we classified some existing defense methods based on their usage stages and characteristics and provided brief introductions along with some advantages and disadvantages. The taxonomies and articles are shown in Fig. 10.

A. Multilevel Backdoor Defense Methods

Following the approach outlined in Liu et al. [117], we categorized defense methods into three levels.

1) Dataset-Level Defense: Tejankar et al. [143] proposed PatchSearch, which quantifies poisoned images based on scores and then iteratively searches for parts of the image with higher scores. These parts are then ranked in descending order using top-k selection. A trained binary classifier is constructed to remove the poisoned samples from the dataset, thus purifying the dataset. Finally, the model is retrained. This method can be used when access to trusted data or image labels is not available, but it is limited to defending against patch-based attacks. Guo et al. [157] employed explainable AI techniques to identify the most important features in samples containing triggers. The purpose was to reduce redundant features that could impact the classification task and achieve trigger pruning. However, for a model with n-class labels, they obtained n pruned trigger masks and patterns. As a result, the authors proposed using mean absolute deviation (MAD) anomaly detection to distinguish the true triggers from this series of restored triggers. They then retrained the model with the recovered trigger samples to achieve trojan removal. However, this method is currently limited to triggers with regular shapes, which imposes certain constraints on its applicability. Li et al. [158] pointed out that applying spatial transformations such as flipping or scaling to test samples can significantly and effectively reduce the success rate of backdoor attacks. However, this method is only applicable to static triggers. Meanwhile, they also highlighted that applying transformations to training samples before training the model can enhance the robustness to trigger variations, which can be used to evade certain defense detection techniques. Udeshi et al. [159] proposed their defense method NEO in a black-box scenario. Specifically, they targeted a more traditional form of attack, where a small pixel block is used as a trigger. NEO uses the dominant colors of the image to cover the trigger, thus defending against backdoor attacks. Although this method has some limitations, it can still provide a certain level of defense and does not require knowledge of the internal details of the model.

Wei et al. [146] used image inpainting algorithms to remove the trigger after they identified the trigger pattern. The poisoned images processed using this method showed better visual results compared to [159]. Additionally, this approach can adapt to triggers of different shapes. As existing backdoor attacks have

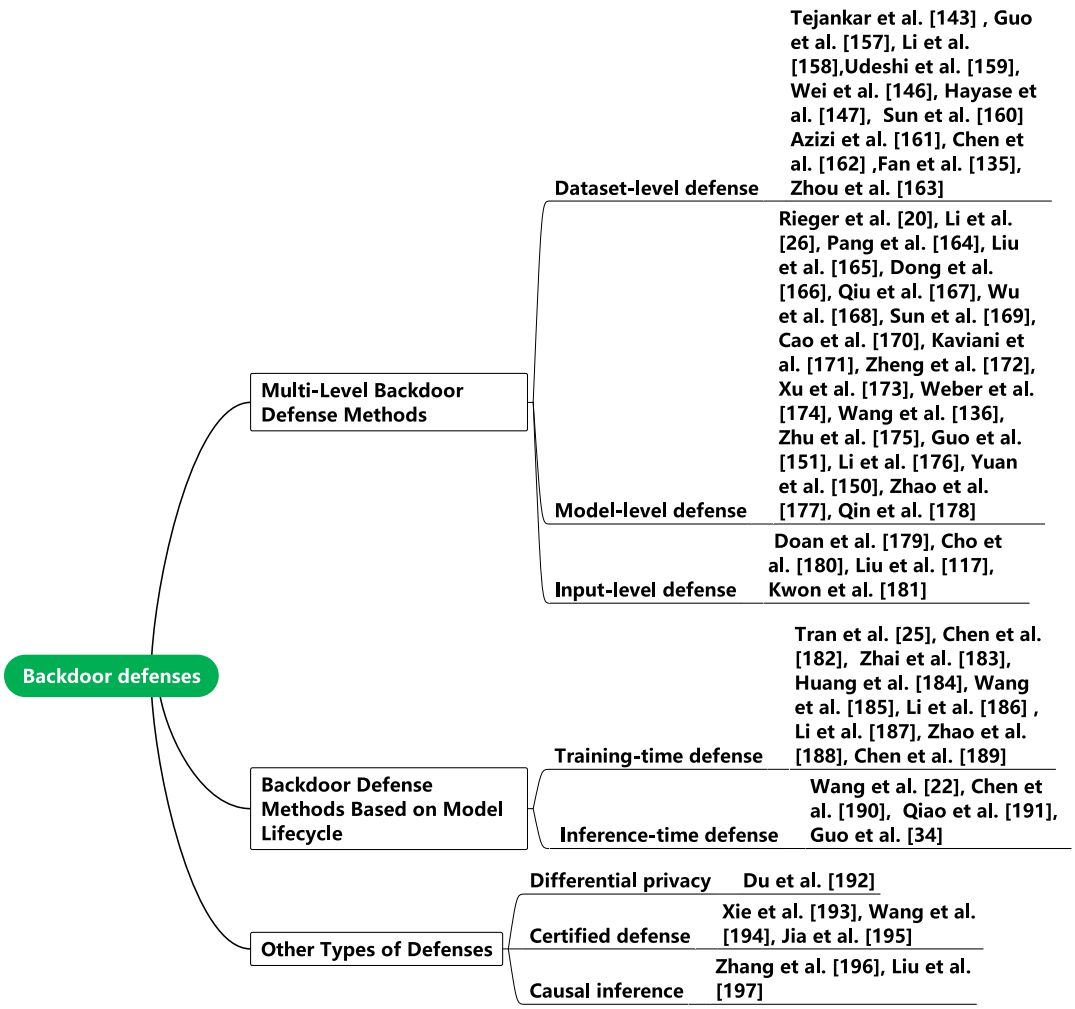

Fig. 10. Taxonomy of backdoor defenses.

attempted to achieve poisoning with the smallest possible sample rate, more advanced defense methods are needed to detect backdoors effectively. As a result, Hayase et al. [147] proposed using covariance estimation to amplify the spectral features of poisoned samples and then applied the quantum entropy score for feature detection, further improving the trigger removal effectiveness. Sun et al. [160] proposed a method called Smooth-Inv, which achieves the recovery of a backdoor trigger with just a single image. This provides a reliable security guarantee for existing classifiers. For text-based backdoor attacks, Azizi et al. [161] introduced T-miner, the first systematic exploration of defending against trojan attacks in the text domain. This approach does not require access to the original dataset; it only relies on training a generative model through synthesized samples. Chen et al. [162] proposed a technique called backdoor keyword identification (BKI), which involves removing samples containing certain keywords from the dataset and then retraining the model with the purified dataset. The selection of keywords is achieved by analyzing changes in the LSTM's internal neurons and scoring the impact of each word in the text. They then choose several words with higher scores from each training sample as the keywords. Since this method requires retraining the model in the end, it may result in significant time overhead. Fan et al. [135] simplified complex RNNs by transforming them into easily understandable models. Then, based on the abstract model, they generated explanation results corresponding to each sentence input. Finally, they detected whether the explanation results belonged to trigger patterns based on one key intuition and mitigated backdoor attacks by removing such patterns. Zhou et al. [163] proposed DATAELIXIR, a novel purification method designed specifically to cleanse poisoned datasets. It utilizes diffusion models to eliminate trigger features and restore benign features, effectively transforming poisoned samples into benign ones.

2) Model-Level Defense: Rieger et al. [20] introduced a novel model filtering technique called DeepSight to filter out model updates with high attack impact. Specifically, they introduced division differences (DDifs), normalized update energies (NEUPs), and the threshold exceedings metric to establish a dynamic filtering mechanism. This mechanism is capable of inferring information about the model training data, effectively identifying and filtering poisoned models. Li et al. [26] presented a method called neural attention distillation (NAD), which utilizes a teacher network to guide a student network with

a backdoor to fine-tune on a small portion of a clean dataset, aligning the attention of the student network's intermediate layers with the teacher network's attention. This effectively eliminates backdoor triggers in deep neural networks. Pang et al. [164] were the first to propose using unlabeled data for defending against backdoors. Specifically, they employed knowledge distillation models for defense. By predicting clean samples, benign knowledge is distilled from the teacher model, enabling the student model to learn the normal behavior of the teacher model, thereby removing backdoors. Furthermore, to optimize the backdoor removal effectiveness, they introduced an adaptive layer-wise initialization strategy to initialize the student model. Liu et al. [165] proposed a defense method that combines pruning and fine-tuning. In essence, pruning is employed to remove the backdoor neurons, while fine-tuning is used to restore (or at least partially recover) the decrease in classification accuracy on clean inputs introduced by pruning. However, this approach necessitates a subset of clean training data, and the computational cost of the fine-tuning component remains substantial. Additionally, this solution could lead to a reduction in the predictive accuracy of clean samples. Dong et al. [166] conducted defense in a more realistic black-box setting, where defenders can only access a pretrained model and use it as an oracle to obtain predictions. Drawing inspiration from natural evolution strategies (NES), they proposed using estimated gradients of parameters to learn a search distribution, aiming to optimize the loss values. Qiu et al. [167] designed the first framework, DeepSweep, for systematically evaluating DNN backdoor attacks. This framework comprises two essential databases: an attack database containing known attack instances and an augmentation database containing common image transformation functions. Both of these databases are extensible, meaning that the framework can continually improve as new attack methods are developed. They then fine-tuned the model's decision boundary and perturbed trigger patterns during the inference stage to defend against backdoor attacks. While this method is tailored for image classification backdoor attacks, it also offers insights for other domains, such as NLP. Wu et al. [168] introduced two algorithms based on neuron input and weights. This method does not require access to the client's raw data, thereby achieving defense against backdoor attacks on the client-side while preserving client data privacy. However, in practical applications, it might incur significant time overhead. Sun et al. [169] reduced the model update norm and added Gaussian noise to alleviate model-replacement-based backdoor attacks, but they did not provide certified robustness guarantees. Cao et al. [170] introduced Integrated FL, the first provably secure FL approach against malicious clients. However, this method requires training hundreds of FL models, which falls short in defending against malicious clients capable of manipulating local training data and model updates. Kaviani et al. [171] combined the scale-free structure and Link-pruning to develop the LSPF algorithm, aimed at eliminating connections between input layer neurons and neurons in other layers. Furthermore, this method is characterized by its simplicity in computation and ease of use. Zheng et al. [172] introduced the first effective data-free defense method against backdoor attacks. They identified a relationship between the Lipschitz constant and backdoor behavior: channels are sensitive to anomalous perturbations, and this constant is often used to measure such sensitivity. Consequently, channels associated with backdoors exhibit higher Lipschitz constants. Therefore, these channels can be pruned to defend against backdoor attacks. Additionally, since this method does not require data, it can be extended to different CNN architectures. Xu et al. [173] trained a meta-classifier to determine whether a model has been implanted with a trojan. Specifically, they approached two challenges. First, they used Jumbo learning to address the issue of providing a training set for classifiers in a black-box mode. Second, building upon the previous approach, they improved detection quality by fine-tuning queries. This method exhibits strong generalization capabilities, but it requires imitating other known trojan attacks to enhance accuracy, incurring significant computational costs. Weber et al. [174] proposed a unified smoothing framework to prove robustness against various attacks, but this method comes with substantial runtime costs. Wang et al. [136] applied specific optimization bounds to each neuron to suppress any potential large activations caused by backdoor attacks, without significantly reducing the classifier's accuracy on clean samples. Zhu et al. [175] proposed ADFL, a novel FL backdoor defense scheme based on adversarial distillation. ADFL generates fake samples containing backdoor features by deploying a generative adversarial network (GAN) on the server side and relabeling fake samples to obtain a distilled dataset. Then, using the labeled samples as input, knowledge distillation is performed with the clean model as the teacher and the global model as the student, correcting the global model and eliminating the influence of backdoor neurons. This effectively defends against backdoor attacks while maintaining model performance. Guo et al. [151] proposed the global adversarial peak (GAP) metric, calculated through the extreme value analysis of adversarial perturbations. The GAP score is ultimately used in conjunction with the median absolute deviation (MAD) to detect backdoors in neural networks. Li et al. [176] discovered that neuron activation patterns show a unique distribution shift in backdoored benign and poisoned data. They introduced layer-wise activation correction (LAC) to align activation distributions between quantized and full-precision models, reducing backdoor neuron drift. They also proposed the poisoned distribution approximation (PDA) objective, using slight perturbations to enhance activation differences and improve defense capabilities. Yuan et al. [150] proposed SHINE, which retrains the shielding agent to learn to take appropriate actions in a poisoned state and maintain its original behavior in a clean state, thereby eliminating the trigger's impact on the backdoor agent. Zhao et al. [177] proposed a model unlearning-based approach, unlearning-based model ablation (UMA), which filters out nonbackdoor features by eliminating inherent features of the target class from the model and then reveals the backdoor through dynamic trigger optimization. Qin et al. [178] proposed a novel anti-backdoor FL framework called Snowball. It selects model updates through bidirectional elections from an individual perspective. The framework introduces a new paradigm for detecting infected models using variational autoencoders

(VAE), which gradually expands the selection scope, focusing on model differences rather than benign patterns themselves to better distinguish between infected and benign model updates.

3) Input-Level Defense: Doan et al. [179] introduced a plug-and-play backdoor defense system called Februus. In essence, this system targets poisoned samples given to the model, removes the trigger, and then repairs the image. This is the first method that defends against backdoors at runtime, enabling the cleaning of trojan inputs without the need to retrain the model or label data. Although Cho et al. [180] is a defense method designed to counter adversarial attacks in semantic segmentation tasks, it has also been used for backdoor defense testing by the authors in [117], so we include it here. In essence, they constructed a denoising autoencoder comprising an encoder and a decoder. The encoder extracts features, while the decoder reconstructs the image based on these features. By preprocessing data using this autoencoder, harmful content is removed from the samples, achieving the defense objective. A similar method is found in [181]. In [181], if the classification result remains unchanged after the sample is processed by a detrigger autoencoder, the model is considered clean; otherwise, it is deemed to have a backdoor.

B. Backdoor Defense Methods Based on Model Lifecycle

1) Training-Time Defense: Tran et al. [25] discovered spectral features as novel characteristics of all known backdoor attacks and proposed a method for detecting backdoor attacks by identifying spectral features. This method requires access to the training dataset of the DNN model, which is not a realistic assumption as defenders might not always have access to the training dataset. Chen et al. [182] first introduced a method called "Activation Clustering," which does not require any trusted data to detect poisoned samples. This technique is applicable to both text and image datasets. However, at the initial stage, this method requires training the model with an untrusted dataset, meaning that a complete and trustworthy dataset for training is not available, making it impractical. Zhai et al. [183] introduced the noise-augmented contrastive learning (NCL) framework. Specifically, it utilizes noise augmentation to generate a new dataset and then retrains the model using the NCL loss function to weaken the association between triggers and target labels. This is the first text defense framework that achieves model cleansing rather than trigger detection. Huang et al. [184] decoupled the model training process by dividing the DNN into two parts: a feature extractor and a simple classifier. They first trained the complete DNN model using an unlabeled dataset. Then, they kept the feature extractor unchanged and trained the remaining fully connected layers using a labeled dataset. Next, they used a label-noise algorithm to assess the credibility of samples. High-credible samples were considered as labeled samples (i.e., most likely clean samples), while low-credible samples (i.e., most likely backdoor samples) were discarded. Finally, the model was fine-tuned using semisupervised learning. Wang et al. [185] introduced nonlinearity (NONE) to identify linear decision regions. The authors observed that trojans in DNNs were always paired with hyperplanes as their trojan regions, and trojan-related neurons formed a hyperplane as the classification surface for the input domain of all affected labels. To address this, they proposed the NONE algorithm to reset the affected neurons and remove the corresponding data samples to enforce nonlinear decision regions, thereby defending against trojan attacks during training. This is also the first method to defend against natural trojans. Li et al. [186] observed two specific differences between backdoor samples and clean samples during model training: 1) backdoor tasks are easier to learn, and the more powerful the backdoor attack is, the easier it is for the model to learn it; 2) backdoor tasks are correlated with the backdoor target class. Therefore, they proposed the antibackdoor learning (ABL) method, which employs a gradient ascent-based anti-backdoor mechanism to defend against backdoor attacks during the model training phase. Li et al. [187] proposed a novel defense method called reconstructive neuron pruning (RNP), which exposes and prunes backdoor neurons through a process of unlearning and then recovery. Specifically, RNP first forgets the neurons by maximizing the model's error on a small set of clean samples, and then recovers the neurons by minimizing the model's error on the same data. Zhao et al. [188] assumed that the end-users only have a clean dataset for fine-tuning, and they mitigate the backdoor impact by pruning the head and further normalizing the weights of the remaining attention heads. Chen et al. [189] proposed a novel and effective defense method called progressive isolation of poisoned data (PIPD). This method gradually isolates poisoned data to improve isolation accuracy and reduce the risk of benign samples being mistakenly classified as poisoned.

2) Inference-Time Defense: Wang et al. [22] proposed the first defense method against DNN backdoor attacks, which can detect and reverse engineer hidden triggers embedded inside deep neural networks. However, this method cannot handle models with multiple classes. Besides, it requires a clean training dataset. Therefore, Chen et al. [190] proposed an approach that requires less prior knowledge of the model and does not rely on the original clean dataset. Specifically, they first used model inversion techniques to obtain a substitution training set. Then, they used a conditional generator to construct triggers and finally employed hypothesis testing-based anomaly detection to detect backdoors. Additionally, the conditional generator can also be used to patch the backdoored model. Qiao et al. [191] found that modeling backdoor triggers poses a high-dimensional unsampled generation modeling problem, where the exact trigger generation process cannot be known. Therefore, they proposed a max-entropy staircase approximator (MESA) algorithm. This algorithm integrates a set of submodels to approximate the unknown trigger distribution. However, this method requires retraining the model and can only defend against nonstructured triggers with fixed shapes and sizes. Guo et al. [34] transformed the trojan detection problem into an optimization problem. Compared to [22], they proposed some new methods. First, they designed new regularization terms to address issues of oversized or scattered extracted triggers and to constrain trigger smoothness. Second, they devised new metrics to better eliminate false positives and incorrect triggers.

Experimental results showed that [34] outperformed [22], but due to the large number of hyperparameters in this method, finding the optimal configuration could be challenging and computationally expensive.

C. Other Types of Backdoor Defenses

1) Differential Privacy: Du et al. [192] demonstrated that backdoor attacks can be seen as injecting poisoned samples into the training dataset, with these poisoned samples being treated as outliers. The authors showcased in their paper that differential privacy can enhance the effectiveness of outlier detection without explicitly defining the outliers, thereby making it capable of detecting various anomalies. Consequently, they extended the application of differential privacy to backdoor defense and achieved favorable outcomes.

2) Certified Defense: Xie et al. [193] proposed the first general framework, certifiably robust dederated Learning (CRFL), for training FL models with certifiable robustness against backdoor attacks. In essence, CRFL employs clipping and smoothing of model parameters to control model smoothness and generates sample-level robustness certification to counter backdoor attacks. However, these methods did not take into account privacy issues in FL and incurred significant losses in FL performance. Wang et al. [194] extended the application of the proof of robustness using random smoothing techniques for adversarial examples to the realm of backdoor defense. This marks the first certified defense against backdoor attacks, although the efficacy of existing random smoothing methods is limited. Jia et al. [195] demonstrated that the inherent majority voting mechanisms of k nearest neighbors (kNN) and radius nearest neighbors (rNN) can provide certified defense against backdoor attacks. They further indicated that jointly certifying multiple test examples yields improved rNN certification robustness guarantees.

3) Causal Inference: Zhang et al. [196] innovatively approached the issue of backdoor defense from a causal inference perspective. They constructed a causal graph and identified false associations introduced by backdoor attacks between input images and target labels, thereby misleading the model's predictions. Building upon this, they introduced causality inspired backdoor defense (CBD), a method to learn deconfounded representations to defend against backdoor attacks. Liu et al. [197] proposed a novel defense framework called front-door adjustment for backdoor elimination (FABE), which is based on causal inference and does not rely on assumptions about the form of the trigger. By creating a "front-door" that maps the actual causal relationships, which refers to the text retaining the semantic equivalence of the initial input and is generated by an additional, fine-tuned language model known as the defense model, FABE effectively distinguishes between spurious and legitimate associations.

VI. Architectures, Datasets, and Metrics

In this section, we summarize the common model architectures, datasets, and evaluation metrics in backdoor research. We created a comprehensive table that includes experimental data from some papers, as shown in Table III, allowing readers to quickly reference relevant information.

A. Architectures

Following the classification of deep learning model architectures in [198], we add the Transformer architecture and divided deep learning model structures into four categories: DNN, CNN, RNN, and Transformer. We provide a brief introduction to each type of architecture, including some model variants, and highlighted backdoor research based on them.

1) DNN: *DNN* plays a crucial role in the field of natural language processing and are widely used in tasks such as toxic content detection [199], fake news detection [200], and neural machine translation [201]. However, pretrained DNNs have provided attackers with opportunities to implant malicious backdoors. Currently, there is a substantial amount of backdoor research based on DNNs, such as composite attack [101], T-miner [161], and TABOR [157].

2) CNN: *CNNs* are widely used in the field of image processing, and there are a large number of backdoor works based on them, such as Badnets [8], ULPs [13], and BaFFLe [128]. Among them, Badnets [8] is a pioneering work in backdoor attacks, where the essence of the attack lies in establishing a connection between triggers and target labels during CNN training. CNNs include architectures such as VGG, ResNet, and more.

VGG [202] was proposed by a research team at the University of Oxford, and its full name is "Visual Geometry Group." The main characteristics of the VGG network are its depth and simplicity. Many backdoor attack articles have used the VGG model [55], [82], [98], [99].

ResNet (residual neural network) is a deep convolutional neural network architecture proposed by researchers from Microsoft [203]. ResNet constructs an extremely deep network by stacking multiple residual blocks, making training easier and improving accuracy. With the widespread use of ResNet, a considerable amount of research on backdoors based on this architecture has begun to emerge [57], [100], [103], [191].

3) RNN: *LSTM* (long short-term memory) is a deep learning model used for processing sequential data. It is widely used in various domains such as natural language processing, speech recognition, and time series prediction. For instance, Dai et al. [116] proposed a black-box backdoor attack on a text classification system based on LSTM. The dynamic placement of triggers was introduced to enhance the evasion capability of the attack to some extent. Additionally, [44], [124], [161], among others, are also conducting backdoor research based on LSTM.

4) Transformer: *Transformer* is a model proposed by Google in 2017 for machine translation [204]. It is essentially an encoder–decoder architecture that can solve various tasks, such as sentiment analysis, machine translation, text summarization, and semantic relationship extraction. Many researchers have studied backdoor based on it [12], [96], [99], [161], [205].

TABLE III
SUMMARY OF ARCHITECTURES, DATASETS, AND METRICS

Dataset	Reference	Architectures	Metrics	Attack Perform. (%)		Defence Perform. (%)	
				ASR	ACC	ASR	ACC
MNIST	Kolouri et al. [13]	CNN	AUC	100	99.4	-	-
	Wang et al. [22]	2 Conv + 2 Dense	ASR, ACC	99.9	98.54	-	-
	Nguyen el al. [123]	3 ConvBlocks, 2 fcs	ASR, ACC	99.86	99.52	-	-
	Xie et al. [37]	2 conv and 2 fc	ASR, ACC	100	99.8	-	-
	Yao et al. [47]	CNN	ASR, ACC	96.6	97.3	-	-
	Nguyen et al. [100]	2 conv, 2 fc	ASR, ACC	99.54	99.54	-	-
	Chen et al. [52]	2 conv, 2 pool and 2 fc	ASR, ACC	99.85	98.2	-	-
	Chen et al. [190]	CNN	ASR, ACC	-	-	7.4	99.1
	Wang et al. [185]	ResNet18	ASR, ACC	99.98	99.61	0.37	99.57
	Guo et al. [157]	DNN	Fidelity, Correctness, Patching Performance, ASR, ACC	100	-	11.2	94.8
	Du et al. [192]	CNN	AUPR, AUROC, ASR, ACC	98.1	99.11	0.39	97.34
	Wei et al. [146]	LeNet5	ASR, ACC	94.8	98.9	1	98
	Zhu et al. [175]	CNN	Relabeling accuracy, ASR, ACC	100	90.39	3.25	88.56
CIFAR10	Yuan et al. [12]	ViT	ASR, ACC	100	94.17	-	-
	Kolour et al. [13]	VGG	AUC, ASR, ACC	99.9	79.5	-	-
	Li et al. [26]	WideResNet	ASR, ACC	-	-	4.77	81.17
	Nguyen et al. [123]	PreActRes18	ASR, ACC	99.55	94.15	-	-
	Nguyen et al. [100]	PreActRes18	ASR, ACC	99.32	94.65	-	-
	Lin et al. [101]	4Conv+3FC	ASR, ACC	80.8	82.4	-	-
	Zhang et al. [196]	WideResNet (WRN-16-1)	ASR, ACC	-	-	1.6	87
	Pang et al. [164]	ResNet-18	ASR, ACC	-	-	3.74	91.59
	Guo et al. [84]	ResNet	AUROC, ASR, ACC	97.7	92.31	-	93.9
	Liu et al. [117]	AlexNet	ASR, ACC	99.9	63.50 ± 0.90	-	-
	Chen et al. [52]	Resnet50	ASR, ACC	95.21	78.25	-	-
	Doan et al. [179]	6 Conv + 2 Dense	ASR, ACC	100	90.79	0.25	90.08
	Chen et al. [145]	ResNet-18	ASR, ACC	97.29	92.46	0.31	92.42
	Huang et al. [184]	Resnet-18	ASR, ACC	100	94.92	0.96	92.41
	Wang et al. [185]	ResNet18	ASR, ACC	100	94.1	1.07	93.62
	Qiu et al. [167]	ResNet18	ASR, ACC	100	83	4.5	78.5
	Li et al. [158]	VGG-19	ASR, ACC	100	91.9	1.6	87.6
	Li et al. [158]	ResNet-34	ASR, ACC	100	94.1	1.5	91.4
	Wang, et al. [136]	ResNet-18	ASR, ACC	99.9	91.6	1.8	90.6
	Zhu et al. [175]	CNN	Relabeling accuracy	93.96	72.54	2.31	71.69
	Li et al. [187]	ResNet-18	ASR, ACC	96.34	93.54	5.03	92.18
CIFAR100	Chen et al. [145]	ResNet-18	ASR, ACC	99.77	74.43	0.07	73.36
	Wang et al. [136]	VGG-16	ASR, ACC	99.8	66.2	1.5	65
GTSRB	Wang et al. [22]	6 Conv + 2 Dense	ASR, ACC	97.4	96.51	-	-
	Li et al. [26]	WideResNet	ASR, ACC	-	-	3.18	80.73
	Liu et al. [57]	ResNet34	SSIM, PSNR, MSE	91.67	86.3	-	-
	Nguyen et al. [123]	PreActRes18	ASR, ACC	98.78	98.97	-	-
	Nguyen et al. [100]	PreActRes18	ASR, ACC	99.84	99.27	-	-
	Lin et al. [101]	6Conv	ASR, ACC	85.6	94	-	-
	Zhang et al. [196]	WideResNet (WRN-16-1)	ASR, ACC	-	-	1.82	95.17
	Pang et al. [164]	ResNet-18	ASR, ACC	-	-	0.54	97.05
	Yao et al. [47]	6 Conv + 2 FC	ASR, ACC	100	85.6	-	-
	Zhang et al. [99]	VGGNet	F1-score, ASR, ACC	100	99.9	-	-
	Kolouri et al. [13]	ResNet	AUC, ASR, ACC	97.7	98.1	-	-
	Chen et al. [190]	Faster-RCNN	ASR, ACC	-	-	8.8	97.1
	Doan et al. [179]	7 Conv + 2 Dense	ASR, ACC	100	96.78	0	96.64
	Wang et al. [185]	ResNet18	ASR, ACC	99.84	96.67	0.76	96.39
	Qiu et al. [167]	LeNet-8	ASR, ACC	67	87.5	2	76.5
	Guo et al. [157]	DNN	Fidelity, Correctness, Patching Performance, ASR, ACC	99.9	-	1.5	94.6
	Wang et al. [136]	MobileNet	ASR, ACC	100	94.5	1.5	95.4

(Continued)

TABLE III
(Continued.) SUMMARY OF ARCHITECTURES, DATASETS, AND METRICS

Dataset	Reference	Architectures	Metrics	Attack Perform.(%)		Defence Perform. (%)	
				ASR	ACC	ASR	ACC
ImageNet	Liu et al. [57]	ResNet34	SSIM, PSNR, MSE, ASR, ACC	82.11	90.32	-	-
	Zhang et al. [196]	ResNet-34	ASR, ACC	-	-	0.91	87.9
	Huang et al. [184]	Resnet-18	ASR, ACC	90.49	80.23	0.26	80.99
	Wang et al. [185]	ResNet18	ASR, ACC	99.07	89.83	0.08	86.34
	Guo et al. [157]	DNN	Fidelity, Correctness, Patching Performance, ASR, ACC	99.9	-	5.8	91.1
	Li et al. [187]	ResNet-34	ASR, ACC	93.89	88.98	8.87	85.91
Tiny-ImageNet	Kolouri et al. [13]	ResNet	AUC, ASR, ACC	99.2	45.1	-	-
	Guo et al. [84]	ResNet	AUROC, ASR, ACC	97.22	40.11	-	90.90
CelebA	Nguyen et al. [123]	ResNet18	ASR, ACC	99.33	78.99	-	-
	Zhu et al. [175]	ResNet18	Relabeling accuracy, ASR, ACC	92.37	71.42	3.97	71.06
YouTube Face	Lin et al. [101]	13Conv+3FC	ASR, ACC	86.3	99.7	-	-
	Guo et al. [157]	DNN	Fidelity, Correctness, Patching Performance, ASR, ACC	99.4	-	4.8	91.7
LFW	Guo et al. [157]	DNN	Fidelity, Correctness, Patching Performance, ASR, ACC	96	-	8.6	91.6
AG's News	Lin et al. [101]	4LSTM+1FC	ASR, ACC	89.2	88.5	-	-
	Azizi et al. [161]	Bi-LSTM	ASR, ACC	99.78 $\pm$ 0.58	90.65 $\pm$ 0.13	-	-
	Zhai et al. [183]	BERT	ASR, ACC	96.49	92.63	44.91	91.02
	Fan et al. [135]	LSTM	ASR, ACC	99.99	90.88	-	-
	Fan et al. [135]	GRU	ASR, ACC	100	90.05	-	-
VGG-Face	Yqo et al. [47]	VGG-Face (13 Conv + 3 FC)	ASR, ACC	100	91.8	-	-
	Doan et al. [179]	13 Conv + 3 Dense (VGG-16)	ASR, ACC	100	91.86	0	91.78
SST	Zhang et al. [99]	BERT	F1-score, ASR, ACC	100	93.2	-	-
	Yang et al. [64]	bert-base-uncased	ASR, ACC	100	92.55	-	-
	Zhai et al. [183]	BERT	ASR, ACC	96.71	90.54	55.81	90.3
OLID	Zhang et al. [99]	BERT	F1-score, ASR, ACC	99.9	80.7	-	-
Enron	Zhang et al. [99]	BERT	F1-score, ASR, ACC	99.2	98.7	-	-
	Wei et al. [68]	BERT-base-cased	ASR, ACC	80.34	98.57	6.21	98.43
cats-VS-dogs	Zhang et al. [99]	VGGNet	F1-score, ASR, ACC	100	96.1	-	-
waste classification	Zhang et al. [99]	VGGNet	F1-score, ASR, ACC	100	92.6	-	-
IMDB	Yang et al. [64]	bert-base-uncased	ASR, ACC	98.74	93.57	-	-
	Chen et al. [162]	Bi-directional LSTM	ASR, ACC	98.9	86.32	13.23	87.03
	Fan et al. [135]	LSTM	ASR, ACC	99.98	90.6	-	-
	Fan et al. [135]	GRU	ASR, ACC	100	89.79	-	-
Amazon	Yang et al. [64]	bert-base-uncased	ASR, ACC	100	97	-	-
Fashion-MNIST	Liu et al. [117]	AlexNet	ASR, ACC	100	86.80 $\pm$ 1.30	-	-
STL10	Liu et al. [117]	AlexNet	ASR, ACC	99.9	43.80 $\pm$ 1.00	-	-
SVHN	Liu et al. [117]	AlexNet	ASR, ACC	97.00 $\pm$ 0.90	75.90 $\pm$ 1.00	-	-
	Chen et al. [52]	Resnet50	ASR, ACC	97.72	93.81	-	-
Yelp	Azizi et al. [161]	3 LSTM	ASR, ACC	99.52 $\pm$ 0.55	92.70 $\pm$ 0.26	-	-
Hate Speech	Azizi et al. [161]	LSTM-based model	ASR, ACC	99.57 $\pm$ 0.11	94.86 $\pm$ 0.24	-	-
Movie Review	Azizi et al. [161]	3 LSTM	ASR, ACC	97.82 $\pm$ 0.13	83.39 $\pm$ 0.44	-	-
	Fan et al. [135]	LSTM	ASR, ACC	99.93	77.47	-	-
	Fan et al. [135]	GRU	ASR, ACC	99.66	76.63	-	-
Fakeddit	Azizi et al. [161]	Transformed-based model using 2-head self-attention layers	ASR, ACC	99.76 $\pm$ 0.03	83.07 $\pm$ 0.09	-	-
BTSR	Doan et al. [179]	ResNet18	ASR, ACC	100	97.04	0.12	96.98
TrojAI	Wang et al. [185]	ResNet18	ASR, ACC	54.33	99.91	34.98	98.13
PubFig	Qui et al. [167]	VGG-16	ASR, ACC	100	95.5	1.5	87
TC	Fan et al. [135]	LSTM	ASR, ACC	99	92.32	-	-
	Fan et al. [135]	GRU	ASR, ACC	100	92.14	-	-
ISIC-2019	Feng et al. [59]	ResNet50	ASR, ACC	99.53 $\pm$ 0.08	85.43 $\pm$ 0.40	-	-

BERT, which stands for bidirectional encoder representation from transformers, is a pretrained language representation model proposed by Google in 2018 [206]. It is one of the most popular NLP models and can achieve state-of-the-art performance in various downstream tasks. Consequently, there is an abundance of research on backdoors based on it, such as [11], [63], [65], [207].

B. Datasets

In Table III, we summarize the datasets used in the referenced papers in this article and classified them. The data in the table is mainly divided into two categories: image data and text data. Each dataset includes corresponding attack and/or defense performance. The table can provide some references for selecting datasets in future research. Next, we categorize the dataset into image and text and provide a brief description of each dataset.

1) Image Dataset:

a) *MNIST:* The MNIST dataset consists of 10 handwritten digits (0–9) presented in grayscale images, with each digit represented by 6000 training images and 1000 testing images. There are lots of works based on it [13], [22], [37], [47], [123].

b) *CIFAR10:* The CIFAR-10 dataset includes 60 000 images spread across ten distinct classes. For each class, the training set contains 5000 images, and the test set has 1000 images. Each image in the dataset is 32 × 32 pixels in size. Many backdoor research utilize it [12], [13], [26], [123], [100].

c) *CIFAR100:* CIFAR-100 dataset contains 100 classes. Each class has 600 color images of size 32 × 32, with 500 images used for training and 100 images for testing. [145] and [136] used this dataset.

d) *GTSRB:* The GTSRB dataset contains 39 209 training images and 12 630 testing images, divided into 43 classes. Each image in the GTSRB dataset is 32 × 32 pixels in size. The related works include [22], [26], [57], [123].

e) *ImageNet:* ImageNet is a dataset containing over 14 million images, covering around 22 000 categories. Each category has hundreds to thousands of images, and the dataset is widely used for image classification and object detection tasks. Many works have used it [57], [196], [184].

f) *Tiny-ImageNet:* Tiny-ImageNet is a smaller subset of the broader ImageNet dataset, consisting of 100 000 training samples and 10 000 testing samples across 200 classes. The images in Tiny-ImageNet are 64 × 64 pixels in size [13], [84], among others, are conducting research based on it.

g) *CelebFaces attributes (Celeba):* The CelebFaces Attributes Dataset, also known as CelebA, is a large-scale dataset of facial attributes. It includes 10 177 identities and 202 599 facial images. Each image is annotated with 5 landmark locations and 40 binary attributes. Research efforts on this dataset include works like [123] and [175].

h) *YouTube Face:* The YouTube Faces Dataset is a facial video database used for studying unconstrained face recognition in videos. It contains 3425 video clips from 1595 subjects, all sourced from YouTube. Numerous studies have utilized the dataset [101], [157].

i) *Labeled Faces in the Wild (LFW):* The LFW dataset contains 13 233 facial images, each labeled with the corresponding person's name, covering 5749 individuals. Most people are represented by only one image. Each image is 250 × 250 pixels in size, with the majority being color images. [157] utilizes this dataset.

j) *VGG-Face:* The VGG-face dataset is a collection of celebrity images scraped from the web, containing images of 2622 celebrities. This dataset is designed to have no overlap with popular face recognition benchmark datasets. The dataset has been employed in a wide range of research [47], [179].

k) *Cats-VS-Dogs:* The Cats versus Dogs dataset contains 25 000 images of cats and dogs, with an equal number of images for each class (12 500 cats and 12 500 dogs). The dataset is often used to train models to distinguish between images of cats and dogs. The dataset is employed in [99].

l) *Waste classification:* The waste classification dataset contains 15 000 images (each 256 × 256 pixels) covering 30 different categories of various recyclable materials, general waste, and household items. Each category has 500 images, with 250 images per subcategory. [99] makes use of the dataset.

m) *Fashion-MNIST:* The Fashion-MNIST dataset includes 70 000 front images of various items from 10 different categories. The size, format, and training/test split of Fashion-MNIST are identical to the original MNIST dataset, with a 60 000/10 000 split for training and testing data, and 28 x 28 grayscale images. [117] leverages the dataset.

n) *STL10:* The STL-10 dataset, derived from ImageNet, contains 10 categories with 500 labeled training samples per category and an additional 100 000 unlabeled samples. Each category has 800 96 x 96 pixel RGB images in the test set. The dataset is incorporated into [117].

o) *SVHN:* The SVHN dataset contains 10 categories, where digits 1–9 correspond to labels 1–9, and the digit "0" is labeled as 10. The training set includes 73 257 images, and the test set contains 26 032 images. Many researchers have made use of the dataset [52], [117].

p) *Belgium traffic sign recognition (BTSR):* The Belgium traffic sign recognition (BTSR) dataset is a widely used, high-resolution traffic sign dataset featuring images of size 224 × 224. Unlike other datasets, BTSR offers a more limited number of training samples. [179] relies on the dataset.

q) *TrojAI:* TrojAI contains the images created by compositing a synthetic traffic sign, with a random background image from the KITTI dataset [208].

r) *PubFig:* The PubFig dataset contains 11 070 training images and 2768 testing images of 83 celebrities. [167] is based on the dataset.

s) *ISIC-2019:* The ISIC-2019 dataset includes 25 331 dermoscopic images categorized into eight diagnostic groups: melanoma, melanocytic nevus, basal cell carcinoma, actinic keratosis, benign keratosis, dermatofibroma, vascular lesion, and squamous cell carcinoma. [59] is conducted using the dataset.

2) *Text Dataset:*

a) *AG's news:* The AG's News dataset is created by selecting the four largest categories from the original corpus: World, Sports, Business, and Sci/Tech. The dataset has seen extensive use in various studies [101], [161], [183].

b) *SST:* SST stands for Stanford Sentiment Treebank, and the data mainly comes from movie reviews. The dataset is divided into train/dev/test sets, containing 67 359 873, and 1822 samples, respectively. A significant number of works have applied the dataset [64], [99], [183].

c) *OLID:* OLID is a hierarchical dataset used for identifying the type and target of offensive language in social media. The dataset was collected on Twitter and is publicly available. It contains a total of 14 100 tweets, with 13 240 in the training set and 860 in the test set. Each tweet is labeled at three levels:1) offensive/nonoffensive; 2) targeted insult/untargeted; and 3) individual/group/other. [99] relies on the dataset.

d) *Enron:* The Enron Email Dataset includes 5 million email messages from 150 employees, consisting of executives and mid-level managers at the Enron Corporation. Several studies have leveraged the dataset [68], [99].

e) *IMDB:* The IMDB dataset consists of 50 000 movie reviews gathered from online sources, with an equal split of 25 000 positive and 25 000 negative reviews. Each review averages around 220 words. The dataset has been frequently used in academic research [64], [135], [162].

f) *Amazon:* The Amazon Product Reviews dataset, collected in 2023, is a large-scale dataset featuring 48.19 million products and 571.54 million reviews from 54.51 million users. The analysis in [64] is performed using the dataset.

g) *Yelp:* The Yelp dataset includes 4.7 million user reviews, over 150 000 business listings, 200 000 images, and data from 12 major metropolitan areas. It also covers 1 million tips from 1.1 million users and more than 1.2 million business attributes. [161] incorporates data from the dataset.

h) *Hate speech:* The hate speech dataset merges two tweet datasets from previous studies [209], [210], each with a different labeling scheme. The first dataset categorizes tweets into two classes: offensive and nonoffensive, while the second categorizes them into three classes: sexist, racist, and neither. [161] makes use of the dataset.

i) *Movie review:* Rotten tomatoes movie reviews (MR) is a dataset of movie reviews that includes 5331 positive and 5331 negative sentences, with an average sentence length of 19 words. Many works have incorporated the dataset into their analysis [135], [161].

j) *Fakeddit:* Fakeddit is a novel multimodal dataset designed for fake news detection, consisting of over 1 million samples of fake news across multiple categories. After several stages of review and processing, the samples are labeled using distant supervision into two-way, three-way, and six-way classification categories. [161] utilizes this dataset.

k) *Toxic comment (TC):* The toxic comment classification (TC) dataset was created for a competition and includes a large collection of Wikipedia comments that have been manually labeled for toxic behavior. [135] relies on the dataset.

C. Metrics

Here, we describe some of the used evaluation metrics in backdoor attacks and defenses.

1) *Attack success rate (ASR):* This metric is used to evaluate the proportion of successful attacks in all of the attacker's trials. Specifically, it measures the rate at which samples containing triggers are misclassified as the targeted type by the backdoor model. Let T denote attack trials, T_s denote the number of successful attacks, and T_f denote the number of failed attacks, then, ASR can be represented using (6) as follows:

$$\text{ASR} = \frac{T_s}{T_s + T_f} \tag{6}$$

2) *Main task accuracy (ACC):* ACC represents the model's performance on clean samples.

3) *Area under the curve (AUC):* AUC is the area under the ROC curve (receiver operating characteristic curve) formed with the coordinate axes. The value of AUC ranges between 0.5 and 1. The closer the AUC is to 1.0, the higher the accuracy of the detection method; when it equals 0.5, the accuracy is the lowest and of no practical use.

4) *Fidelity [157]:* Similar to the F1-score, it is also used to quantify the fidelity of backdoor recovery. Given a restored trigger, the F_1 score is defined as following: $F_1 = 2 \cdot (\text{precision} \cdot \text{recall}/\text{precision} + \text{recall})$, $\text{precision} = (\|\mathbf{M} \odot \mathbf{M}_t\|_1/\|\mathbf{M}\|_1)$, $\text{recall} = (\|\mathbf{M} \odot \mathbf{M}_t\|_1/\|\mathbf{M}_t\|_1)$. Here, M and M_t represent the mask of the trigger restored and that of the ground-truth trigger.

5) *Correctness [157]:* Correctness includes four different detection outcomes: successful detection, successful detection with errors, incorrect trojan detection, and failed detection.

6) *Patching performance [157]:* Patching performance is assessed using classification accuracy and attack success rate on the patched model. Classification accuracy measures the ratio of contaminated testing samples that are correctly classified into their original classes, while attack success rate measures the ratio of contaminated testing samples that are misclassified into the target classes.

7) *Area under the precision-recall curve (AUPR):* AUPR is the area under the Precision-Recall curve and typically better reflects the model's performance when the dataset is imbalanced.

8) *Relabeling accuracy [175]:* Relabeling accuracy represents the ratio of the number of generated samples

correctly classified by the clean model to the total number of generated samples.

9) *Structural similarity (SSIM):* SSIM is a metric for measuring the similarity between two images. The SSIM algorithm is mainly used to detect the similarity of two images of the same size or to assess the degree of image distortion. It is commonly used to detect the similarity between backdoor samples and normal samples to achieve covert trigger implantation.
10) *Peak signal to noise ratio (PSNR):* PSNR is a commonly used full-reference image quality assessment metric. The basic definition of PSNR is based on mean square error (MSE), which measures the difference between the original image and the noisy image. Specifically, the larger the PSNR, the smaller the distortion and the better the image quality.
11) *Mean squared error (MSE):* MSE loss, also known as quadratic loss or L2 loss, is commonly used in regression tasks. The MSE function measures the performance of a model by calculating the square of the distance (i.e., error) between the predicted values and the actual values. That is, the closer the predicted values are to the true values, the smaller the mean squared error.
12) *F1-score:* F1 Score is a metric used in statistics to measure the accuracy of a binary classification model. The F1 Score can be seen as the harmonic mean of precision and recall, and is often used to assess the accuracy of an algorithm. Its maximum value is 1, and its minimum value is 0.

VII. Future Research Direction

As mentioned above, backdoors have been extensively studied in multiple fields from various perspectives. However, there are still some research problems that need to be addressed in certain areas. Therefore, in this section, we propose several potential research directions, hoping to provide some guidance for future backdoor research.

1) *Innovative triggers design exploration:* Trigger forms vary widely, including but not limited to static and dynamic triggers, visible and invisible triggers, pixel and text triggers, and more. Different tasks and scenarios require different triggers, making the design of sufficiently covert and effective triggers a crucial research point in backdoor research.
2) *Backdoor mechanisms exploration:* Traditional backdoor implantation typically involves the joint participation of datasets and models. Recent research has introduced training-free backdoor attacks [93] and data-free backdoor attacks [96]. In [93], backdoors are implanted by directly modifying model components, while [96] implants backdoors by generating substitutable datasets. Both of these methods break away from the inherent patterns of backdoor attacks and achieve high ASR. Therefore, further investigating the mechanisms of backdoor attacks to propose more advanced attack methods is a promising research direction.
3) *The positive utilization:* Technology is a double-edged sword. Although backdoor attacks have been shown to be harmful, researchers have taken advantage of their stealthiness to use backdoor attacks in data protection scenarios, including copyright protection [211], [212], [213]. For instance, in [211], the authors watermark datasets using poison-only backdoor attacks, and then they confirm dataset ownership using a hypothesis testing technique. This approach allows ownership verification with only model API access. Thus, future research may examine more beneficial applications of backdoor methods.
4) *Expanding research areas and application scenarios:* We visualized the indexed keywords related to backdoor attacks from 2018 to 14 July, 2024 in the Scopus database using the VOSviewer tool, as shown in Fig. 11. In the image, each node represents an index keyword. The size of the node indicates the frequency or importance of the keyword in the research field. The entire graph centers on "backdoor" and uses different colors to represent different clusters. Specifically, the green cluster is mainly related to "backdoor attack" and "deep neural networks," the red cluster to "network security" and "malware," the yellow cluster to machine learning security, the blue cluster to NLP and "neural networks," and the purple cluster to more specialized topics such as "backdoor sets" and "algorithms." Additionally, the thickness of the connections between nodes reflects the strength of their relationships. Finally, the peripheral nodes, although on the edge, still hold significant research value in the field, such as "face recognition," "privacy-preserving techniques," and "hardware." This visualization offers a detailed overview of the interconnections between various research topics, emphasizing the interdisciplinary nature of the field and pinpointing key areas of interest and study. However, there is limited research in areas such as speaker recognition [17], [114], [115], video recognition [112], [113], [124], 3-D point clouds [111], [118], and large language models [109], [110]. In addition, Fig. 12 illustrates the publication trends of relevant papers on backdoor research over the past five years. The data in Fig. 12 also comes from the Scopus database. We only counted the number of articles in the field of Computer Science that have "backdoor" in the title, abstract, or keywords, with data up to 14 July, 2024. As shown in the figure, the number of published articles on backdoors has been increasing year by year, indicating that this field continues to hold significant research value. Through the analysis above, we aim to showcase the trends in backdoor research and use it as a basis to further enrich and deepen research on various aspects of backdoors.
5) *Interpretability of backdoor attacks:* The interpretability of backdoor attacks remains an unresolved issue. Some studies have attempted to elucidate the interpretability of backdoor attacks [91], [157], [214], [215], [216]. Specifically, Li et al. [216] found that the success of backdoor poisoning attacks depends on three key components: the

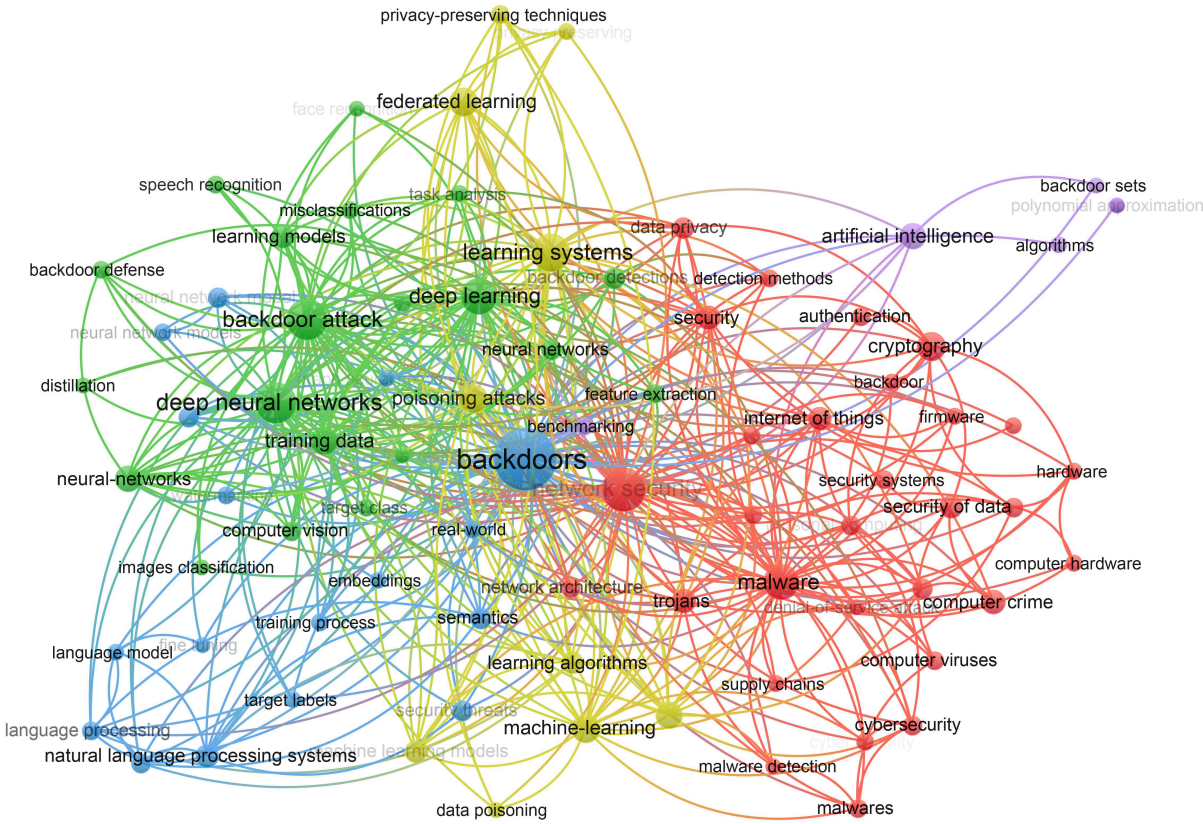

Fig. 11. Research index keyword network.

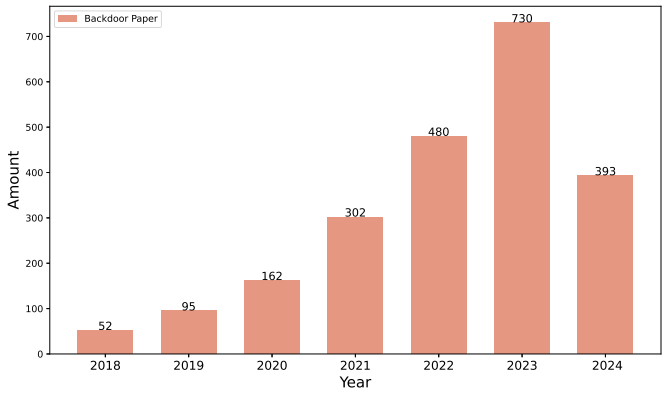

Fig. 12. Published articles statistics.

number of feature vectors in the dataset, the trigger pattern and the norm ratio of the feature vectors, and the percentage of poisoned data in the training set. Increasing research into the interpretability of backdoors is crucial for developing more advanced attacks and more effective defenses.

6) *Persistence of backdoor attacks:* Dai et al. [108] found that the backdoors implanted by current attacks are not persistent and disappear quickly once the attacker stops poisoning the model. Therefore, they proposed a novel attack, Chameleon, which leverages contrastive learning to further amplify the attack effect and create more persistent backdoors. Additionally, [104], [105], [106], [107] conducted relevant research on the persistence of backdoor attacks. Exploring the reasons for extending the attack's effect will help better understand the backdoor mechanisms and design more covert and persistent attack methods. At the same time, this provides an important theoretical foundation and practical guidance for developing more advanced and effective defense strategies.

VIII. Conclusion

Although deep learning systems are extensively used in computer vision, natural language processing, and speech recognition settings, it has been discovered that these systems are susceptible to backdoor attacks. Sorting out the present status of attack and defense is therefore essential. In this article, we systematically discuss deep learning backdoor attacks and defenses. Specifically, we summarized and categorized backdoor attacks with deep learning approaches, applications, attacker's knowledge and other criteria. Furthermore, we classified backdoor detection methods based on timing of detection and detection object. In addition, we categorized backdoor defenses according to multiple levels, the lifecycle of the model, and other relevant factors. Next, we conclude by discussing a few more exciting potential research directions. According to the investigated literature, the capabilities of backdoor attacks and defenses are dynamically intertwined and evolving. As a result, backdoor attacks and defenses continue to be hot topics of discussion. We anticipate that our work will be a useful resource

for scholars studying this topic and that it will spark additional conversations about the development of deep learning backdoor attacks and defenses.

REFERENCES

[1] S. Dong, P. Wang, and K. Abbas, "A survey on deep learning and its applications," *Comput. Sci. Rev.*, vol. 40, 2021, Art. no. 100379.

[2] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016.

[3] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 2, pp. 604–624, Feb. 2021.

[4] Z. Zhang, J. Geiger, J. Pohjalainen, A. E.-D. Mousa, W. Jin, and B. Schuller, "Deep learning for environmentally robust speech recognition: An overview of recent developments," *ACM Trans. Intell. Syst. Technol.*, vol. 9, no. 5, pp. 1–28, 2018.

[5] M. Wang and W. Deng, "Deep face recognition: A survey," *Neurocomputing*, vol. 429, pp. 215–244, Mar. 2021.

[6] A. Yadav and D. K. Vishwakarma, "Sentiment analysis using deep learning architectures: A review," *Artif. Intell. Rev.*, vol. 53, no. 6, pp. 4335–4385, 2020.

[7] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3523–3542, Jul. 2022.

[8] T. Gu, B. Dolan-Gavitt, and S. Garg, "BadNets: Identifying vulnerabilities in the machine learning model supply chain," 2017, *arXiv:1708.06733*.

[9] S. Li et al., "Hidden backdoors in human-centric language models," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2021, pp. 3123–3140.

[10] X. Zhang, Z. Zhang, S. Ji, and T. Wang, "Trojaning language models for fun and profit," in *Proc. IEEE Eur. Symp. Secur. Privacy (EuroS & P),* Piscataway, NJ, USA: IEEE Press, 2021, pp. 179–197.

[11] X. Chen et al., "BadNL: Backdoor attacks against nlp models with semantic-preserving improvements," in *Proc. Annu. Comput. Secur. Appl. Conf.*, 2021, pp. 554–569.

[12] Z. Yuan, P. Zhou, K. Zou, and Y. Cheng, "You are catching my attention: Are vision transformers bad learners under backdoor attacks?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 24605–24615.

[13] 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.*, 2020, pp. 301–310.

[14] C. Liao, H. Zhong, A. Squicciarini, S. Zhu, and D. Miller, "Backdoor embedding in convolutional neural network models via invisible perturbation," 2018, *arXiv:1808.10307*.

[15] J. Ye, X. Liu, Z. You, G. Li, and B. Liu, "DriNet: Dynamic backdoor attack against automatic speech recognization models," *Appl. Sci.*, vol. 12, no. 12, 2022, Art. no. 5786.

[16] Y. Kong and J. Zhang, "Adversarial audio: A new information hiding method and backdoor for dnDNN-based speech recognition models," 2019, *arXiv:1904.03829*.

[17] 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.*, Toronto, ON, Canada. Piscataway, NJ, USA: IEEE Press, 2021, pp. 2560–2564.

[18] S. Koffas, J. Xu, M. Conti, and S. Picek, "Can you hear it? Backdoor attacks via ultrasonic triggers," in *Proc. ACM Workshop Wireless Secur. Mach. Learn.*, 2022, pp. 57–62.

[19] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *Proc. Int. Conf. Artif. Intell. Statist.*. PMLR, 2020, pp. 2938–2948.

[20] P. Rieger, T. D. Nguyen, M. Miettinen, and A.-R. Sadeghi, "Deepsight: Mitigating backdoor attacks in federated learning through deep model inspection," 2022, *arXiv:2201.00763*.

[21] H. Wang et al., "Attack of the tails: Yes, you really can backdoor federated learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 16070–16084.

[22] B. Wang et al., "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *Proc. IEEE Symp. Secur. Privacy (SP),* Piscataway, NJ, USA: IEEE Press, 2019, pp. 707–723.

[23] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "STRIP: A defence against trojan attacks on deep neural networks," in *Proc. 35th Annu. Comput. Secur. Appl. Conf.*, 2019, pp. 113–125.

[24] Y. Liu, W.-C. Lee, G. Tao, S. Ma, Y. Aafer, and X. Zhang, "ABS: Scanning neural networks for back-doors by artificial brain stimulation," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2019, pp. 1265–1282.

[25] B. Tran, J. Li, and A. Madry, "Spectral signatures in backdoor attacks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, *arXiv:1811.00636*.

[26] Y. Li, X. Lyu, N. Koren, L. Lyu, B. Li, and X. Ma, "Neural attention distillation: Erasing backdoor triggers from deep neural networks," 2021, *arXiv:2101.05930*.

[27] X. Sheng, Z. Han, P. Li, and X. Chang, "A survey on backdoor attack and defense in natural language processing," in *Proc. IEEE 22nd Int. Conf. Softw. Qual., Rel. Secur. (QRS),* Piscataway, NJ, USA: IEEE Press, 2022, pp. 809–820.

[28] M. Omar, "Backdoor learning for NLP: Recent advances, challenges, and future research directions," 2023, *arXiv:2302.06801*.

[29] Y. Li, S. Zhang, W. Wang, and H. Song, "Backdoor attacks to deep learning models and countermeasures: A survey," *IEEE Open J. Comput. Soc.*, vol. 4, pp. 134-146, 2023.

[30] Y. Gao et al., "Backdoor attacks and countermeasures on deep learning: A comprehensive review," 2020, *arXiv:2007.10760*.

[31] W. Guo, B. Tondi, and M. Barni, "An overview of backdoor attacks against deep neural networks and possible defences," *IEEE Open J. Signal Process.*, vol. 3, pp. 261-287, 2022.

[32] Y. Li, Y. Jiang, Z. Li, and S.-T. Xia, "Backdoor learning: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 1, pp. 5-22, Jan. 2024.

[33] G. Cui, L. Yuan, B. He, Y. Chen, Z. Liu, and M. Sun, "A unified evaluation of textual backdoor learning: Frameworks and benchmarks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 5009–5023.

[34] W. Guo, L. Wang, X. Xing, M. Du, and D. Song, "TABOR: A highly accurate approach to inspecting and restoring trojan backdoors in AI systems," 2019, *arXiv:1908.01763*.

[35] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.,* PMLR, 2017, pp. 1273–1282.

[36] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, "Analyzing federated learning through an adversarial lens," in *Proc. Int. Conf. Mach. Learn.,* PMLR, 2019, pp. 634–643.

[37] C. Xie, K. Huang, P.-Y. Chen, and B. Li, "DBA: Distributed backdoor attacks against federated learning," in *Proc. Int. Conf. Learn. Representations*, 2019.

[38] P. Chen, J. Yang, J. Lin, Z. Lu, Q. Duan, and H. Chai, "A practical clean-label backdoor attack with limited information in vertical federated learning," in *Proc. IEEE Int. Conf. Data Mining (ICDM),* Piscataway, NJ, USA: IEEE Press, 2023, pp. 41–50.

[39] M. Naseri, Y. Han, and E. De Cristofaro, "BADVFL: Backdoor attacks in vertical federated learning," 2023, *arXiv:2304.08847*.

[40] H. Zhuang, M. Yu, H. Wang, Y. Hua, J. Li, and X. Yuan, "Backdoor federated learning by poisoning backdoor-critical layers," 2023, *arXiv:2308.04466*.

[41] S. S. Mousavi, M. Schukat, and E. Howley, "Deep reinforcement learning: An overview," in *Proc. SAI Intell. Syst. Conf. (IntelliSys), vol. 2,* Cham, Switzerland: Springer, 2018, pp. 426–440.

[42] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 26–38, Nov. 2017.

[43] J. Cui, Y. Han, Y. Ma, J. Jiao, and J. Zhang, "BadRL: Sparse targeted backdoor attack against reinforcement learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 10, 2024, pp. 11 687–11694.

[44] L. Wang, Z. Javed, X. Wu, W. Guo, X. Xing, and D. Song, "BACKDOORL: Backdoor attack against competitive reinforcement learning," 2021, *arXiv:2105.00579*.

[45] P. Kiourti, K. Wardega, S. Jha, and W. Li, "TrojDRL: Evaluation of backdoor attacks on deep reinforcement learning," in *Proc. 57th ACM/IEEE Des. Automat. Conf.,* Piscataway, NJ, USA: IEEE Press, 2020, pp. 1–6.

[46] Y. Chen, Z. Zheng, and X. Gong, "MARNet: Backdoor attacks against cooperative multi-agent reinforcement learning," *IEEE Trans. Dependable Secure Comput.*, vol. 20, no. 5, pp. 4188–4198, Sep./Oct. 2023.

[47] Y. Yao, H. Li, H. Zheng, and B. Y. Zhao, "Latent backdoor attacks on deep neural networks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2019, pp. 2041–2055.

[48] F. Zhuang et al., "A comprehensive survey on transfer learning," *Proc. IEEE,* vol. 109, no. 1, pp. 43–76, Jan. 2020.

[49] 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.*, vol. 15, no. 3, pp. 1526–1539, May/Jun. 2022.

[50] P. Li, J. Huang, S. Zhang, C. Qi, C. Liang, and Y. Peng, "A novel backdoor attack adapted to transfer learning," in *Proc. IEEE Smartworld, Ubiquitous Intell. & Comput., Scalable Comput. & Commun., Digit. Twin, Privacy Comput., Metaverse, Auton. & Trusted Vehicles (SmartWorld/UIC/ScalCom/DigitalTwin/PriComp/Meta),* Piscataway, NJ, USA: IEEE Press, 2022, pp. 1730–1735.

[51] Y. Matsuo and K. Takemoto, "Backdoor attacks on deep neural networks via transfer learning from natural images," *Appl. Sci.*, vol. 12, no. 24, 2022, Art. no. 12564.

[52] K. Chen, H. Zhang, X. Feng, X. Zhang, B. Mi, and Z. Jin, "Backdoor attacks against distributed swarm learning," *ISA Trans.*, vol. 141, pp. 59–72, Oct. 2023.

[53] S. Warnat-Herresthal et al., "Swarm learning for decentralized and confidential clinical machine learning," *Nature*, vol. 594, no. 7862, pp. 265–270, 2021.

[54] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *Proc. 4th Int. Conf. Comput. Commun. Control Automat.,* Piscataway, NJ, USA: IEEE Press, 2018, pp. 1–6.

[55] 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.*, 2021, pp. 16463–16472.

[56] S. Li, M. Xue, B. Z. H. 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.

[57] Y. Liu, X. Ma, J. Bailey, and F. Lu, "Reflection backdoor: A natural backdoor attack on deep neural networks," in *Proc. Comput. Vis. - ECCV - 16th Eur. Conf.*, Glasgow, UK. Cham, Switzerland: Springer, 2020, pp. 182–199.

[58] A. Turner, D. Tsipras, and A. Madry, "Label-consistent backdoor attacks," 2019, *arXiv:1912.02771*.

[59] Y. Feng, B. Ma, J. Zhang, S. Zhao, Y. Xia, and D. Tao, "Fiba: Frequency-injection based backdoor attack in medical image analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 20876–20885.

[60] M. Nwadike, T. Miyawaki, E. Sarkar, M. Maniatakos, and F. Shamout, "Explainability matters: Backdoor attacks on medical imaging," 2020, *arXiv:2101.00008*.

[61] Y. Matsuo and K. Takemoto, "Backdoor attacks to deep neural network-based system for COVID-19 detection from chest X-ray images," *Appl. Sci.*, vol. 11, no. 20, 2021, Art. no. 9556.

[62] H. Lan, J. Gu, P. Torr, and H. Zhao, "Influencer backdoor attack on semantic segmentation," 2023, *arXiv:2303.12054*.

[63] K. Kurita, P. Michel, and G. Neubig, "Weight poisoning attacks on pre-trained models," 2020, *arXiv:2004.06660*.

[64] W. Yang, L. Li, Z. Zhang, X. Ren, X. Sun, and B. He, "Be careful about poisoned word embeddings: Exploring the vulnerability of the embedding layers in NLP models," 2021, *arXiv:2103.15543*.

[65] H. Kwon and S. Lee, "Textual backdoor attack for the text classification system," *Secur. Commun. Netw.*, vol. 2021, no. 1, pp. 1–11, 2021.

[66] X. Pan, M. Zhang, B. Sheng, J. Zhu, and M. Yang, "Hidden trigger backdoor attack on {NLP} models via linguistic style manipulation," in *Proc. 31st USENIX Secur. Symp. (USENIX Secur. 22)*, 2022, pp. 3611–3628.

[67] J. Li, Y. Yang, Z. Wu, V. Vydiswaran, and C. Xiao, "Chatgpt as an attack tool: Stealthy textual backdoor attack via blackbox generative model trigger," 2023, *arXiv:2304.14475*.

[68] C. Wei et al., "LMSanitator: Defending prompt-tuning against task-agnostic backdoors," 2023, *arXiv:2308.13904*.

[69] S. Zhao, M. Jia, L. A. Tuan, F. Pan, and J. Wen, "Universal vulnerabilities in large language models: Backdoor attacks for in-context learning," 2024, *arXiv:2401.05949*.

[70] Z. Xiang, F. Jiang, Z. Xiong, B. Ramasubramanian, R. Poovendran, and B. Li, "Badchain: Backdoor chain-of-thought prompting for large language models," 2024, *arXiv:2401.12242*.

[71] J. Yan et al., "Backdooring instruction-tuned large language models with virtual prompt injection," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol.* (Vo. 1 Long Papers), 2024, pp. 6065–6086.

[72] L. Struppek, D. Hintersdorf, and K. Kersting, "Rickrolling the artist: Injecting backdoors into text encoders for text-to-image synthesis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 4584–4596.

[73] Y. Li, T. Li, K. Chen, J. Zhang, S. Liu, W. Wang, T. Zhang, and Y. Liu, "Badedit: Backdooring large language models by model editing," 2024, *arXiv:2403.13355*.

[74] Y. Nie et al., "TrojFM: Resource-efficient backdoor attacks against very large foundation models," 2024, *arXiv:2405.16783*.

[75] M. Zhu, S. Wei, H. Zha, and B. Wu, "Neural polarizer: A lightweight and effective backdoor defense via purifying poisoned features," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1132–1153.

[76] E. Hubinger et al., "Sleeper agents: Training deceptive llms that persist through safety training," 2024, *arXiv:2401.05566*.

[77] S. Liang, M. Zhu, A. Liu, B. Wu, X. Cao, and E.-C. Chang, "Badclip: Dual-embedding guided backdoor attack on multimodal contrastive learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 24645–24654.

[78] D. Lu, T. Pang, C. Du, Q. Liu, X. Yang, and M. Lin, "Test-time backdoor attacks on multimodal large language models," 2024, *arXiv:2402.08577*.

[79] J. Liang et al., VL-Trojan: Multimodal instruction backdoor attacks against autoregressive visual language models," 2024, *arXiv:2402.13851*.

[80] H. Cai, P. Zhang, H. Dong, Y. Xiao, S. Koffas, and Y. Li, "Towards stealthy backdoor attacks against speech recognition via elements of sound," *IEEE Trans. Inf. Forensics Secur.*, vol. 19, pp. 5852–5866, 2024.

[81] S.-H. Chan, Y. Dong, J. Zhu, X. Zhang, and J. Zhou, "BadDet: Backdoor attacks on object detection," in *Proc. Eur. Conf. Comput. Vis*., Cham, Switzerland: Springer, 2022, pp. 396–412.

[82] 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*.

[83] Y. Gao, Y. Li, L. Zhu, D. Wu, Y. Jiang, and S.-T. Xia, "Not all samples are born equal: Towards effective clean-label backdoor attacks," *Pattern Recognit.*, vol. 139, 2023, Art. no. 109512.

[84] J. Guo, Y. Li, X. Chen, H. Guo, L. Sun, and C. Liu, "Scale-up: An efficient black-box input-level backdoor detection via analyzing scaled prediction consistency," 2023, *arXiv:2302.03251*.

[85] A. Saha, A. Subramanya, and H. Pirsiavash, "Hidden trigger backdoor attacks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 07, 2020, pp. 11957–11965.

[86] E. Quiring and K. Rieck, "Backdooring and poisoning neural networks with image-scaling attacks," in *Proc. IEEE Secur. Privacy Workshops (SPW),* Piscataway, NJ, USA: IEEE Press, 2020, pp. 41–47.

[87] M. Barni, K. Kallas, and B. Tondi, "A new backdoor attack in CNNs by training set corruption without label poisoning," in *Proc. IEEE Int. Conf. Image Process.,* Piscataway, NJ, USA: IEEE Press, 2019, pp. 101–105.

[88] E. Dai, M. Lin, X. Zhang, and S. Wang, "Unnoticeable backdoor attacks on graph neural networks," in *Proc. ACM Web Conf.* , 2023, pp. 2263–2273.

[89] E. Wenger, J. Passananti, A. N. Bhagoji, Y. Yao, H. Zheng, and B. Y. Zhao, "Backdoor attacks against deep learning systems in the physical world," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6206–6215.

[90] J. Jia, Y. Liu, and N. Z. Gong, "Badencoder: Backdoor attacks to pre-trained encoders in self-supervised learning," in *Proc. IEEE Symp. Secur. Privacy (SP),* Piscataway, NJ, USA: IEEE Press, 2022, pp. 2043–2059.

[91] H. Zheng, H. Xiong, J. Chen, H. Ma, and G. Huang, "Motif-backdoor: Rethinking the backdoor attack on graph neural networks via motifs," *IEEE Trans. Comput. Social Syst.*, vol. 11, no. 2, pp. 2479–2493, Apr. 2024.

[92] Y. Yu, Y. Wang, W. Yang, S. Lu, Y.-P. Tan, and A. C. Kot, "Backdoor attacks against deep image compression via adaptive frequency trigger," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 12250–12259.

[93] Y. Huang, T. Y. Zhuo, Q. Xu, H. Hu, X. Yuan, and C. Chen, "Training-free lexical backdoor attacks on language models," in *Proc. ACM Web Conf.*, 2023, pp. 2198–2208.

[94] Y. Liu et al., "Trojaning attack on neural networks," in *Proc. 25th Annu. Netw. Distrib. System Secur. Symp. (NDSS),* San Diego, CA, USA: Internet Soc, 2018.

[95] Z. Xi, R. Pang, S. Ji, and T. Wang, "Graph backdoor," in *Proc. 30th USENIX Secur. Symp. (USENIX Secur. 21)*, 2021, pp. 1523–1540.

[96] P. Lv, C. Yue, R. Liang, Y. Yang, S. Zhang, H. Ma, and K. Chen, "A data-free backdoor injection approach in neural networks," in *32nd USENIX Secur. Symp. (USENIX Secur. 23)*, 2023, pp. 2671–2688.

[97] K. Doan, Y. Lao, and P. Li, "Backdoor attack with imperceptible input and latent modification," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 18944–18957.

[98] A. Salem, R. Wen, M. Backes, S. Ma, and Y. Zhang, "Dynamic backdoor attacks against machine learning models," in *Proc. IEEE 7th Eur. Symp. Secur. Privacy (EuroS & P),* Piscataway, NJ, USA: IEEE Press, 2022, pp. 703–718.

[99] Z. Zhang et al., "Red alarm for pre-trained models: Universal vulnerability to neuron-level backdoor attacks," *Mach. Intell. Res.*, vol. 20, no. 2, pp. 180–193, 2023.

[100] T. A. Nguyen and A. Tran, "Input-aware dynamic backdoor attack," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 3454–3464.

[101] J. Lin, L. Xu, Y. Liu, and X. Zhang, "Composite backdoor attack for deep neural network by mixing existing benign features," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2020, pp. 113–131.

[102] S.-Y. Chou, P.-Y. Chen, and T.-Y. Ho, "How to backdoor diffusion models?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 4015–4024.

[103] K. Doan, Y. Lao, W. Zhao, and P. Li, "Lira: Learnable, imperceptible and robust backdoor attacks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 11966–11976.

[104] M. Alam, E. Sarkar, and M. Maniatakos, "Perdoor: Persistent backdoors in federated learning using adversarial perturbations," in *Proc. IEEE Int. Conf. Omni-layer Intell. Syst. (COINS),* Piscataway, NJ, USA: IEEE Press, 2023, pp. 1–6.

[105] D. Yang, S. Luo, J. Zhou, L. Pan, X. Yang, and J. Xing, "Efficient and persistent backdoor attack by boundary trigger set constructing against federated learning," *Inf. Sci.*, vol. 651, 2023, Art. no. 119743.

[106] F. Yao, "Deepvenom: Persistent dnn backdoors exploiting transient weight perturbations in memories," in *Proc. IEEE Symp. Secur. Privacy (SP),* Los Alamitos, CA, USA: IEEE Comput. Soc. Press, 2024, pp. 244–244.

[107] T. Liu et al., "Beyond traditional threats: A persistent backdoor attack on federated learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 19, 2024, pp. 21359–21367.

[108] Y. Dai and S. Li, "Chameleon: Adapting to peer images for planting durable backdoors in federated learning," in *Proc. Int. Conf. Mach. Learn.,* PMLR, 2023, pp. 6712–6725.

[109] W. Yang, X. Bi, Y. Lin, S. Chen, J. Zhou, and X. Sun, "Watch out for your agents! investigating backdoor threats to llm-based agents," 2024, *arXiv:2402.11208*.

[110] H. Huang, Z. Zhao, M. Backes, Y. Shen, and Y. Zhang, "Composite backdoor attacks against large language models," 2023, *arXiv:2310.07676*.

[111] 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.*, 2021, pp. 7597–7607.

[112] X. Gong, Z. Fang, B. Li, T. Wang, Y. Chen, and Q. Wang, "Palette: Physically-realizable backdoor attacks against video recognition models," *IEEE Trans. Dependable Secure Comput.*, vol. 21, no. 04, pp. 2672–2685, Jul./Aug. 2024.

[113] H. A. Al Kader Hammoud, S. Liu, M. Alkhrashi, F. Albalawi, and B. Ghanem, "Look listen and attack: Backdoor attacks against video action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 3439–3450.

[114] H. Guo, X. Chen, J. Guo, L. Xiao, and Q. Yan, "Masterkey: Practical backdoor attack against speaker verification systems," in *Proc. 29th Annu. Int. Conf. Mobile Comput. Netw.*, 2023, pp. 1–15.

[115] X. Li, J. Ze, C. Yan, Y. Cheng, X. Ji, and W. Xu, "Enrollment-stage backdoor attacks on speaker recognition systems via adversarial ultrasound," *IEEE Internet Things J.*, vol. 11, no. 8, pp. 13108–13124, Apr. 2024.

[116] J. Dai, C. Chen, and Y. Li, "A backdoor attack against lstm-based text classification systems," *IEEE Access*, vol. 7, pp. 138872–138878, 2019.

[117] Y. Liu, Z. Li, M. Backes, Y. Shen, and Y. Zhang, "Backdoor attacks against dataset distillation," 2023, *arXiv:2301.01197*.

[118] L. Fan, F. He, T. Si, W. Tang, and B. Li, "Invisible backdoor attack against 3D point cloud classifier in graph spectral domain," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 19, 2024, pp. 21072–21080.

[119] X. Li, S. Wang, R. Huang, M. Gowda, and G. Kesidis, "Temporal-distributed backdoor attack against video based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 4, 2024, pp. 3199–3207.

[120] Y. Gao, H. Chen, P. Sun, J. Li, A. Zhang, Z. Wang, and W. Liu, "A dual stealthy backdoor: From both spatial and frequency perspectives," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 3, 2024, pp. 1851–1859.

[121] T. Wang, Y. Yao, F. Xu, S. An, H. Tong, and T. Wang, "An invisible black-box backdoor attack through frequency domain," in *Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer, 2022, pp. 396–413.

[122] F. Qi et al., "Hidden killer: Invisible textual backdoor attacks with syntactic trigger," 2021, *arXiv:2105.12400*.

[123] A. Nguyen and A. Tran, "Wanet–imperceptible warping-based backdoor attack," 2021, *arXiv:2102.10369*.

[124] S. Zhao, X. Ma, X. Zheng, J. Bailey, J. Chen, and Y.-G. Jiang, "Clean-label backdoor attacks on video recognition models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14443–14452.

[125] T. Huynh, D. Nguyen, T. Pham, and A. Tran, "Combat: Alternated training for effective clean-label backdoor attacks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 3, 2024, pp. 2436–2444.

[126] L. Yu, S. Liu, Y. Miao, X.-S. Gao, and L. Zhang, "Generalization bound and new algorithm for clean-label backdoor attack," 2024, *arXiv:2406.00588*.

[127] R. Ning, J. Li, C. Xin, and H. Wu, "Invisible poison: A blackbox clean label backdoor attack to deep neural networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM),* Piscataway, NJ, USA: IEEE Press, 2021, pp. 1–10.

[128] S. Andreina, G. A. Marson, H. Möllering, and G. Karame, "Baffle: Backdoor detection via feedback-based federated learning," in *Proc. IEEE 41st Int. Conf. Distrib. Comput. Syst.,* Piscataway, NJ, USA: IEEE Press, 2021, pp. 852–863.

[129] X. Chen, W. Guo, G. Tao, X. Zhang, and D. Song, "Bird: generalizable backdoor detection and removal for deep reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 40786–40798.

[130] Y. Li et al., "NTD: Non-transferability enabled deep learning backdoor detection," *IEEE Trans. Inf. Forensics Secur.*, vol. 19, pp. 104–119, 2024.

[131] H. Fu, P. Krishnamurthy, S. Garg, and F. Khorrami, "Differential analysis of triggers and benign features for black-box dnn backdoor detection," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 4668–4680, 2023.

[132] Y. Gao et al., "Design and evaluation of a multi-domain trojan detection method on deep neural networks," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 4, pp. 2349–2364, Jul./Aug. 2022.

[133] T. Wang, Y. Yao, F. Xu, M. Xu, S. An, and T. Wang, "Inspecting prediction confidence for detecting black-box backdoor attacks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 1, 2024, pp. 274–282.

[134] Y. Liu, G. Shen, G. Tao, Z. Wang, S. Ma, and X. Zhang, "Complex backdoor detection by symmetric feature differencing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15003–15013.

[135] M. Fan, Z. Si, X. Xie, Y. Liu, and T. Liu, "Text backdoor detection using an interpretable rnn abstract model," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 4117–4132, 2021.

[136] H. Wang, Z. Xiang, D. J. Miller, and G. Kesidis, "MM-BD: Post-training detection of backdoor attacks with arbitrary backdoor pattern types using a maximum margin statistic," in *Proc. IEEE Symp. Secur. Privacy (SP),* Los Alamitos, CA, USA: IEEE Comput. Soc. Press, 2023, pp. 15–15.

[137] M. Pan, Y. Zeng, L. Lyu, X. Lin, and R. Jia, "{ASSET}: Robust backdoor data detection across a multiplicity of deep learning paradigms," in *Proc. 32nd USENIX Secur. Symp. (USENIX Secur.)*, 2023, pp. 2725–2742.

[138] M. Xue, Y. Wu, Z. Wu, Y. Zhang, J. Wang, and W. Liu, "Detecting backdoor in deep neural networks via intentional adversarial perturbations," *Inf. Sci.*, vol. 634, pp. 564–577, Jul. 2023.

[139] W. Ma, D. Wang, R. Sun, M. Xue, S. Wen, and Y. Xiang, "The" beatrix" resurrections: Robust backdoor detection via gram matrices," 2022, *arXiv:2209.11715*.

[140] W. Li, P.-Y. Chen, S. Liu, and R. Wang, "PSBD: Prediction shift uncertainty unlocks backdoor detection," 2024, *arXiv:2406.05826*.

[141] Y. Wang, W. Li, M. Alam, M. Maniatakos, and S. E. Jabari, "Backdozer: A backdoor detection methodology for drl-based traffic controllers," *J. Auton. Transport. Syst.*, vol. 1, no. 4, pp. 1–22, 2024.

[142] Z. Guan, M. Hu, S. Li, and A. Vullikanti, "Ufid: A unified framework for input-level backdoor detection on diffusion models," 2024, *arXiv:2404.01101*.

[143] A. Tejankar et al., "Defending against patch-based backdoor attacks on self-supervised learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 12239–12249.

[144] F. Qi, Y. Chen, M. Li, Y. Yao, Z. Liu, and M. Sun, "Onion: A simple and effective defense against textual backdoor attacks," 2020, *arXiv:2011.10369*.

[145] W. Chen, B. Wu, and H. Wang, "Effective backdoor defense by exploiting sensitivity of poisoned samples," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 9727–9737.

[146] Y. Wei, H. Gao, Y. Wang, Y. Gao, and H. Liu, "A lightweight backdoor defense framework based on image inpainting," *Neurocomputing*, vol. 537, pp. 22–36, Jun. 2023.

[147] J. Hayase and W. Kong, "Spectre: Defending against backdoor attacks using robust covariance estimation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 4129–4139.

[148] S. Feng et al., "Detecting backdoors in pre-trained encoders," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 16352–16362.

[149] L. Hou, R. Feng, Z. Hua, W. Luo, L. Y. Zhang, and Y. Li, "IBD-PSC: Input-level backdoor detection via parameter-oriented scaling consistency," 2024, *arXiv:2405.09786*.

[150] Z. Yuan, W. Guo, J. Jia, B. Li, and D. Song, "SHINE: Shielding backdoors in deep reinforcement learning," in *Proc. 41st Int. Conf. Mach. Learn.*, ser. Proc. Mach. Learn. Res., R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, Eds., vol. 235. PMLR, Jul. 2024, pp. 57887–57904. [Online]. Available: https://proceedings.mlr.press/v235/yuan24c.html

[151] J. Guo, A. Li, and C. Liu, "Aeva: Black-box backdoor detection using adversarial extreme value analysis," 2021, *arXiv:2110.14880*.

[152] T. Huster and E. Ekwedike, "Top: Backdoor detection in neural networks via transferability of perturbation," 2021, *arXiv:2103.10274*.

[153] B. Sun, J. Sun, W. Koh, and J. Shi, "Neural network semantic backdoor detection and mitigation: A causality-based approach," in *Proc. 33rd USENIX Secur. Symp.,* San Francisco, CA, USA: USENIX Assoc., 2024, pp. 2883–2900.

[154] S. Huang, W. Peng, Z. Jia, and Z. Tu, "One-pixel signature: Characterizing cnn models for backdoor detection," in *Comput. Vis.–ECCV: 16th Eur. Conf.*, Glasgow, UK. Cham, Switzerland: Springer, 2020, pp. 326–341.

[155] W. Jiang, X. Wen, J. Zhan, X. Wang, Z. Song, and C. Bian, "Critical path-based backdoor detection for deep neural networks," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 35, no. 3, pp. 4032–4046, 2022.

[156] Y. Mo, H. Huang, M. Li, A. Li, and Y. Wang, "TERD: A unified framework for safeguarding diffusion models against backdoors," in *Proc. 41st Int. Conf. Mach. Learn.*, ser. Proc. Mach. Learn. Res., R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, Eds., vol. 235, PMLR, Jul 2024, pp. 35892–35909, [Online]. Available: https://proceedings.mlr.press/v235/mo24a.html

[157] W. Guo, L. Wang, Y. Xu, X. Xing, M. Du, and D. Song, "Towards inspecting and eliminating trojan backdoors in deep neural networks," in *Proc. IEEE Int. Conf. Data Mining,* Piscataway, NJ, USA: IEEE Press, 2020, pp. 162–171.

[158] Y. Li, T. Zhai, B. Wu, Y. Jiang, Z. Li, and S. Xia, "Rethinking the trigger of backdoor attack," 2020, *arXiv:2004.04692*.

[159] S. Udeshi, S. Peng, G. Woo, L. Loh, L. Rawshan, and S. Chattopadhyay, "Model agnostic defence against backdoor attacks in machine learning," *IEEE Trans. Rel.*, vol. 71, no. 2, pp. 880–895, Jun. 2022.

[160] M. Sun and Z. Kolter, "Single image backdoor inversion via robust smoothed classifiers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 8113–8122.

[161] A. Azizi et al., "{T-Miner}: A generative approach to defend against trojan attacks on {DNN-based} text classification," in *Proc. 30th USENIX Secur. Symp. (USENIX Secur. 21)*, 2021, pp. 2255–2272.

[162] C. Chen and J. Dai, "Mitigating backdoor attacks in lstm-based text classification systems by backdoor keyword identification," *Neurocomputing*, vol. 452, pp. 253–262, Sep. 2021.

[163] J. Zhou, P. Lv, Y. Lan, G. Meng, K. Chen, and H. Ma, "Dataelixir: Purifying poisoned dataset to mitigate backdoor attacks via diffusion models," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 19, 2024, pp. 21850–21858.

[164] L. Pang, T. Sun, H. Ling, and C. Chen, "Backdoor cleansing with unlabeled data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 12218–12227.

[165] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *Proc. Int. Symp. Res. Attacks, Intrusions, Defenses*, Cham, Switzerland: Springer, 2018, pp. 273–294.

[166] Y. Dong et al., "Black-box detection of backdoor attacks with limited information and data," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 16482–16491.

[167] H. Qiu, Y. Zeng, S. Guo, T. Zhang, M. Qiu, and B. Thuraisingham, "Deepsweep: An evaluation framework for mitigating dnn backdoor attacks using data augmentation," in *Proc. ACM Asia Conf. Comput. Commun. Secur.*, 2021, pp. 363–377.

[168] C. Wu, X. Yang, S. Zhu, and P. Mitra, "Mitigating backdoor attacks in federated learning," 2020, *arXiv:2011.01767*.

[169] Z. Sun, P. Kairouz, A. T. Suresh, and H. B. McMahan, "Can you really backdoor federated learning?" 2019, *arXiv:1911.07963*.

[170] X. Cao, J. Jia, and N. Z. Gong, "Provably secure federated learning against malicious clients," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 8, 2021, pp. 6885–6893.

[171] S. Kaviani, S. Shamshiri, and I. Sohn, "A defense method against backdoor attacks on neural networks," *Expert Syst. Appl.*, vol. 213, 2023, Art. no. 118990.

[172] R. Zheng, R. Tang, J. Li, and L. Liu, "Data-free backdoor removal based on channel lipschitzness," in *Proc. Eur. Conf. Comput. Vis.,* Springer, 2022, pp. 175–191.

[173] X. Xu, Q. Wang, H. Li, N. Borisov, C. A. Gunter, and B. Li, "Detecting ai trojans using meta neural analysis," in *Proc. IEEE Symp. Secur. Privacy (SP),* Piscataway, NJ, USA: IEEE Press, 2021, pp. 103–120.

[174] M. Weber, X. Xu, B. Karlaš, C. Zhang, and B. Li, "Rab: Provable robustness against backdoor attacks," in *Proc. IEEE Symp. Secur. Privacy (SP),* Piscataway, NJ, USA: IEEE Press, 2023, pp. 1311–1328.

[175] C. Zhu, J. Zhang, X. Sun, B. Chen, and W. Meng, "ADFL: Defending backdoor attacks in federated learning via adversarial distillation," *Comput. & Secur.*, vol. 132, 2023, Art. no. 103366.

[176] B. Li et al., "Purifying quantization-conditioned backdoors via layer-wise activation correction with distribution approximation," in *Proc. 41st Int. Conf. Mach. Learn.*, 2024, pp. 27439–27456.

[177] Y. Zhao, C. Li, and K. Chen, "Uma: Facilitating backdoor scanning via unlearning-based model ablation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 19, 2024, pp. 21823–21831.

[178] Z. Qin, F. Chen, C. Zhi, X. Yan, and S. Deng, "Resisting backdoor attacks in federated learning via bidirectional elections and individual perspective," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 13, 2024, pp. 14677–14685.

[179] B. G. Doan, E. Abbasnejad, and D. C. Ranasinghe, "Februus: Input purification defense against trojan attacks on deep neural network systems," in *Proc. Annu. Comput. Secur. Appl. Conf.*, 2020, pp. 897–912.

[180] S. Cho, T. J. Jun, B. Oh, and D. Kim, "DAPAS: Denoising autoencoder to prevent adversarial attack in semantic segmentation," in *Proc. Int. Joint Conf. Neural Netw.*, Piscataway, NJ, USA: IEEE Press, 2020, pp. 1–8.

[181] H. Kwon, "Defending deep neural networks against backdoor attack by using de-trigger autoencoder," *IEEE Access*, Oct. 18, 2021, doi: 10.1109/ACCESS.2021.3086529.

[182] B. Chen et al., "Detecting backdoor attacks on deep neural networks by activation clustering," 2018, *arXiv:1811.03728*.

[183] S. Zhai et al., "NCL: Textual backdoor defense using noise-augmented contrastive learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.,* Piscataway, NJ, USA: IEEE Press, 2023, pp. 1–5.

[184] K. Huang, Y. Li, B. Wu, Z. Qin, and K. Ren, "Backdoor defense via decoupling the training process," 2022, *arXiv:2202.03423*.

[185] Z. Wang, H. Ding, J. Zhai, and S. Ma, "Training with more confidence: Mitigating injected and natural backdoors during training," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 36396–36410.

[186] Y. Li, X. Lyu, N. Koren, L. Lyu, B. Li, and X. Ma, "Anti-backdoor learning: Training clean models on poisoned data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 14900–14912.

[187] Y. Li et al., "Reconstructive neuron pruning for backdoor defense," in *Proc. Int. Conf. Mach. Learn.,* PMLR, 2023, pp. 19837–19854.

[188] X. Zhao, D. Xu, and S. Yuan, "Defense against backdoor attack on pre-trained language models via head pruning and attention normalization," in *Proc. 41st Int. Conf. Mach. Learn.*, R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, Eds.,

vol. 235, PMLR, Jul. 2024, pp. 61108–61120, [Online]. Available: https://proceedings.mlr.press/v235/zhao24r.html

[189] Y. Chen, H. Wu, and J. Zhou, "Progressive poisoned data isolation for training-time backdoor defense," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 10, 2024, pp. 11425–11433.

[190] H. Chen, C. Fu, J. Zhao, and F. Koushanfar, "DeepInspect: A black-box trojan detection and mitigation framework for deep neural networks." in *Proc. 28th Int. Joint Conf. Arti. Intel. (IJCAI),* vol. 2, no. 5, 2019, Art. no. 8.

[191] X. Qiao, Y. Yang, and H. Li, "Defending neural backdoors via generative distribution modeling," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 14004–14013.

[192] M. Du, R. Jia, and D. Song, "Robust anomaly detection and backdoor attack detection via differential privacy," 2019, *arXiv:1911.07116*.

[193] C. Xie, M. Chen, P.-Y. Chen, and B. Li, "Crfl: Certifiably robust federated learning against backdoor attacks," in *Proc. Int. Conf. Mach. Learn.,* PMLR, 2021, pp. 11372–11382.

[194] B. Wanget al., "On certifying robustness against backdoor attacks via randomized smoothing," 2020, *arXiv:2002.11750*.

[195] J. Jia, Y. Liu, X. Cao, and N. Z. Gong, "Certified robustness of nearest neighbors against data poisoning and backdoor attacks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 9, 2022, pp. 9575–9583.

[196] Z. Zhang, Q. Liu, Z. Wang, Z. Lu, and Q. Hu, "Backdoor defense via deconfounded representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 12228–12238.

[197] Y. Liu, X. Xu, Z. Hou, and Y. Yu, "Causality based front-door defense against backdoor attack on language models," in *Proc. 41st Int. Conf. Mach. Learn.*, R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, Eds., vol. 235, PMLR, Jul. 2024, pp. 32239–32252. [Online]. Available: https://proceedings.mlr.press/v235/liu24bu.html

[198] M. Z. Alom et al., "A state-of-the-art survey on deep learning theory and architectures," *Electronics*, vol. 8, no. 3, 2019, Art. no. 292.

[199] S. V. Georgakopoulos, S. K. Tasoulis, A. G. Vrahatis, and V. P. Plagianakos, "Convolutional neural networks for toxic comment classification," in *Proc. 10th Hellenic Conf. Artif. Intell.*, 2018, pp. 1–6.

[200] N. O'Brien, S. Latessa, G. Evangelopoulos, and X. Boix, "The language of fake news: Opening the black-box of deep learning based detectors," Workshop on "AI for Social Good", NIPS 2018, Montreal, Canada, 2018.

[201] E. Wallace, M. Stern, and D. Song, "Imitation attacks and defenses for black-box machine translation systems," 2020, *arXiv:2004.15015*.

[202] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[203] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[204] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5998–6008.

[205] Y. Wu et al., "Attacking pre-trained recommendation," 2023, *arXiv:2305.03995*.

[206] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[207] Z. Li, D. Mekala, C. Dong, and J. Shang, "BFClass: A backdoor-free text classification framework," 2021, *arXiv:2109.10855*.

[208] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, 2013.

[209] Z. Waseem and D. Hovy, "Hateful symbols or hateful people? Predictive features for hate speech detection on twitter," in *Proc. NAACL student Res. Workshop*, 2016, pp. 88–93.

[210] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proc. Int. AAAI Conf. Web Social Media*, vol. 11, no. 1, 2017, pp. 512–515.

[211] Y. Li, M. Zhu, X. Yang, Y. Jiang, T. Wei, and S.-T. Xia, "Black-box dataset ownership verification via backdoor watermarking," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 2318–2332, 2023.

[212] G. Hua, A. B. J. Teoh, Y. Xiang, and H. Jiang, "Unambiguous and high-fidelity backdoor watermarking for deep neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 8, pp. 11204–11217, Aug. 2024.

[213] X. Liu, F. Li, B. Wen, and Q. Li, "Removing backdoor-based watermarks in neural networks with limited data," in *Proc. 25th Int. Conf. Pattern Recognit.*. Piscataway, NJ, USA: IEEE Press, 2021, pp. 10149–10156.

[214] X. Huang, M. Alzantot, and M. Srivastava, "Neuroninspect: Detecting backdoors in neural networks via output explanations," 2019, *arXiv:1911.07399*.

[215] K. Zhang et al., "Exploring the orthogonality and linearity of backdoor attacks," in *Proc. IEEE Symp. Secur. Privacy (SP),* Los Alamitos, CA, USA: IEEE Comput. Soc. Press, 2024, pp. 225–225.

[216] B. Li and W. Liu, "A theoretical analysis of backdoor poisoning attacks in convolutional neural networks," in *Proc. 41st Int. Conf. Mach. Learn.*, R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, Eds., vol. 235. PMLR, Jul. 2024, pp. 28309–28342. [Online]. Available: https://proceedings.mlr.press/v235/li24at.html