BackdoorMBTI: A Backdoor Learning Multimodal Benchmark Tool Kit for Backdoor Defense Evaluation

Haiyang Yu[*]
haiyang_yu@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, Min Hang, China

Tian Xie[*]
xietian1164567053@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, Min Hang, China

Jiaping Gui[*][†]
jgui@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, Min Hang, China

Pengyang Wang
pywang@um.edu.mo
University of Macau
Taipa, Macau SAR, China

Pengzhou Cheng
cpztsm520@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, Min Hang, China

Ping Yi
yiping@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, Min Hang, China

Yue Wu[†]
wuyue@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, Min Hang, China

Abstract

Over the past few years, the emergence of backdoor attacks has presented significant challenges to deep learning systems, allowing attackers to insert backdoors into neural networks. When data with a trigger is processed by a backdoor model, it can lead to mispredictions targeted by attackers, whereas normal data yields regular results. The scope of backdoor attacks is expanding beyond computer vision and encroaching into areas such as natural language processing and speech recognition. Nevertheless, existing backdoor defense methods are typically tailored to specific data modalities, restricting their application in multimodal contexts. While multimodal learning proves highly applicable in facial recognition, sentiment analysis, action recognition, visual question answering, the security of these models remains a crucial concern. Specifically, there are no existing backdoor benchmarks targeting multimodal applications or related tasks.

In order to facilitate the research in multimodal backdoor, we introduce BackdoorMBTI, the first backdoor learning toolkit and benchmark designed for multimodal evaluation across three representative modalities from eleven commonly used datasets. BackdoorMBTI provides a systematic backdoor learning pipeline, encompassing data processing, data poisoning, backdoor training, and evaluation. The generated poison datasets and backdoor models enable detailed evaluation of backdoor defense methods. Given the diversity of modalities, BackdoorMBTI facilitates systematic evaluation across different data types. Furthermore, BackdoorMBTI offers a standardized approach to handling practical factors in backdoor learning, such as issues related to data quality and erroneous labels. We anticipate that BackdoorMBTI will expedite future research in backdoor defense methods within a multimodal context. Code is available at https://anonymous.4open.science/r/BackdoorMBTI-D6A1/README.md.

CCS Concepts

• **Computing methodologies → Artificial intelligence**; • **Security and privacy**;

Keywords

data poisoning, backdoor attack, backdoor defense, multimodal evaluation

ACM Reference Format:
Haiyang Yu, Tian Xie, Jiaping Gui, Pengyang Wang, Pengzhou Cheng, Ping Yi, and Yue Wu. 2018. BackdoorMBTI: A Backdoor Learning Multimodal Benchmark Tool Kit for Backdoor Defense Evaluation. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX).* ACM, New York, NY, USA, 12 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

With the advancement and widespread use of artificial intelligence (AI), neural networks have become an integral component of our modern life, handling diverse data from various devices and applications. However, they face growing threats from backdoor attacks, which are rapidly evolving and present real-world risks. Users may encounter poison data, where attackers infiltrate specific triggers into datasets before training, potentially impacting all users of these compromised datasets. As neural networks scale up, training costs also increase, forcing users to rely on third-party training resources that may lack security, thereby highlighting the real threat posed by backdoor attacks in practical scenarios. To counter the impact of

[*] Both authors contributed equally to this research.
[†] corresponding authors

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

backdoor attacks, researchers have proposed various countermeasures, including backdoor detection methods [5, 7, 26, 48], model repair techniques [36, 40, 80, 89], and so on. In recent years, several backdoor learning benchmarks [2, 37, 53, 79] have also been developed. However, they follow the same path as the defense work and mainly focus on one specific domain.

As current efforts are primarily concentrated on designing efficient algorithms within specific domains such as computer vision, there is a pressing need for modality support. However, the existing benchmarks and defense methods are practically inadequate to meet such a need, primarily due to the diversity of modal data many real-world applications involve. This problem is caused by multiple factors. Firstly, the complexity of real-world data makes it challenging to conduct research and evaluate defense methods effectively in multimodal areas, resulting in uncertainties about their effectiveness. Secondly, the absence of a standardized evaluation baseline makes it hard to provide an objective evaluation of different algorithms. Lastly, current solutions often overlook empirical factors such as noise, undermining their overall efficacy in the real world. Nonetheless, addressing these issues is challenging. This is driven by the inherent complexity of the migration task, particularly within the framework of a multimodal application that necessitates support for diverse modalities and models. Additionally, attacks and defenses may involve numerous private parameters in their settings, making it difficult to evaluate them under a standard baseline.

To address the above issues, we introduce BackdoorMBTI, a novel and unified benchmark and toolkit, dedicated to the evaluation of multimodal backdoor learning. Our benchmark comprises three key aspects: 1) BackdoorMBTI supports three modalities and incorporates eleven representative datasets, seventeen attacks, and seven defense methods. It encompasses diverse classification task scenarios and extends attacks beyond computer vision to include audio and text domains. Furthermore, we adapt the defense methods to a multimodal context and release an open-source backdoor learning benchmark for image, text, and audio backdoor learning. 2) BackdoorMBTI provides easy-to-access backdoor datasets and establishes a standard benchmark for evaluating backdoor defenses in multimodal settings. 3) BackdoorMBTI takes into account factors such as low-quality data and erroneous labels, which align with real-world scenarios.

BackdoorMBTI offers a unified pipeline that ensures fair evaluation within a multimodal context, distinguishing it from other benchmarks. While existing backdoor learning benchmarks primarily focus on unimodal tasks, especially in computer vision, multimodal backdoor learning remains largely unexplored despite the ubiquity of diverse modal data in real-world applications. Table 1 illustrates the modality support in current benchmarks. We can see that only BackdoorBench [79] and Backdoor101 [2] support both image and text. However, the two modalities implemented in these two benchmarks are integrated separately. To the best of our knowledge, we are the first to design a benchmark encompassing all three (i.e., image, text, and audio) backdoor learning modalities.

In our experiments, BackdoorMBTI has demonstrated the ability to effectively handle the aforementioned issues. We have also incorporated a noise generator to simulate noises from both data and labels in the real world. Our experimental findings suggest that, while noises may not significantly impact the performance of backdoor attacks, they play a crucial role in backdoor defenses.

Table 1: The modality support in current benchmarks.

Benchmarks	Image	Text	Audio
TrojAI [27]	✓		
TrojanZoo [53]	✓		
BackdoorBench [79]	✓	✓	
BackdoorBox [37]	✓		
OpenBackdoor [12]		✓	
Backdoor101 [2]	✓	✓	
Ours	✓	✓	✓

We summarize our contributions as follows:

- BackdoorMBTI integrated eleven datasets, seventeen attacks, and seven defenses, covering diverse application scenarios, such as object classification, facial recognition, sentiment analysis, speech command recognition, and speaker identification.
- To enhance reproducibility and facilitate multimodal backdoor research, BackdoorMBTI offers an open-source backdoor learning framework that supports image, text, and audio tasks. We have developed a unified evaluation pipeline that is both user-friendly and easily extensible. Additionally, BackdoorMBTI provides the community with accessible poisoned datasets and models for defense evaluation. The source codes are available at https://anonymous.4open.science/r/BackdoorMBTI-D6A1/README.md.
- To improve practicality, we include noise factors as a robustness assessment module, simulating real-world backdoor defense applications with low-quality data and erroneous labels. Our findings, from a multimodal and empirical perspective, indicate that noise factors enhance model robustness, thereby improving defense performance.

2 Related Work

2.1 Backdoor Attack

Existing backdoor attacks are typically categorized into three types, data poisoning attacks, training control attacks, and model modification attacks [21, 33]. Data poisoning attacks involve the adversary manipulating the training data only [4, 9, 19, 34, 43, 44, 51, 64, 86], while training control attacks allow the adversary to not only manipulate the data but also control the training process [2, 15, 39, 52, 84]. Model-modified attacks, as described in [3, 56, 67], enable the adversary to manipulate the model directly.

Backdoor attack research has primarily focused on computer vision. However, in recent years, this research direction has broadened its scope, extending beyond images to include text and audio [12, 18, 23]. To support these emerging types of backdoor attacks, we design BackdoorMBTI to work in a multimodal paradigm.

2.2 Backdoor Defense

The existing defense methods can be categorized according to the machine learning lifecycle as follows:

Table 2: An overview of 11 datasets included in BackdoorMBTI.

Task	Dataset	Modality	Classes	Total Instances	Used In Experiment
Object Classification	CIFAR10 [30]	Image	10	60,000	[8, 15, 20, 24, 25, 35, 36, 38, 42, 51, 52, 57, 59, 61, 62, 64, 66, 68–71, 75–77, 80, 81, 85, 86, 88–91]
	TinyImageNet [14, 31]	Image	200	110,000	[2, 8, 15, 22, 25, 34, 35, 42, 44, 47, 57, 59, 62, 64, 76, 89, 90]
Traffic Sign Recognition	GTSRB [65]	Image	43	51,839	[4, 7, 15, 17, 22, 26, 35, 36, 42, 44, 47, 51, 52, 57, 61, 66, 68, 74–77, 84–86, 91]
Facial Recognition	CelebA [46]	Image	8*	202,599	[47, 51, 77]
Sentiment Analysis	SST-2 [63]	Text	2	11,855	[12, 54]
	IMDb [49]	Text	2	50,000	[2, 6, 12, 17, 47, 83]
Topic Classification	DBpedia [1]	Text	14	630,000	[6]
	AG's News [87]	Text	4	31,900	[12, 54]
Speech Command Recognition	SpeechCommands [78]	Audio	35	105,829	[17, 40, 81]
Music Genre Classification	GTZAN [72]	Audio	10	1,000	-
Speaker Identification	VoxCeleb1 [50]	Audio	1251	100,000	-

*CelebA is a face attributes classification dataset, and all the face attributes are labeled in a bin format, in order to do the multi-class classification task on it, we follow the same settings in [77].

Preprocessing. The goal of defenders is to destroy backdoor patterns on instances. This is motivated by the observation that backdoor attacks may lose their efficacy when the trigger used during the attack differs from the one used during the poisoning process, ShrinkPad [38] follows the paradigm to prevent backdoor triggering. In the study by [45], an autoencoder is employed as a preprocessor defined between the input and the neural network to address this issue. Additionally, in Deepsweep [58], researchers explored 71 data augmentation techniques to counter backdoor attacks.

Poison suppression. The goal of defenders is to prevent backdoor injections during the training process. For instance, ABL [35] accomplishes poisoning suppression through a two-step approach, which is backdoor sample identification and unlearning. In DBD [25], Huang et al. prevent the clustering observed in backdoor injection via self-supervised learning and semi-supervised fine-tuning. NONE [75] proposed a training approach to prevent the generation of hyperplane for backdoors based on the observation backdoor neurons create a hyperplane in the affected labels through piecewise linear functions.

Backdoor removing and mitigation. The objective of the defenders is to eliminate or reduce the impact of backdoor effects in backdoor models, typically achieved through fine-tuning and pruning. In [40], the effectiveness of fine-tuning (FT) and fine-pruning (FP) in defending against backdoor attacks is demonstrated. Consequently, significant research has focused on how to identify backdoor neurons, which is crucial for pruning-based methods. ANP [80] identifies backdoor neurons using adversarial perturbations, CLP [89] employs channel-based Lipschitz sensitivity, DBR [8] uses the sensitivity metric feature consistency towards transformations, while I-BAU [85] formulates it as a minimax problem to unlearn triggers. MCR [88] enhances model robustness using mode connectivity, SEAM [91] utilizes catastrophic forgetting to erase backdoor triggers, SFT [61] demonstrates the effectiveness of fine-tuning in eliminating most advanced backdoors, and NAD [36] employs attention distillation.

Backdoor detection. There are two primary categories for detecting backdoor attacks in machine learning models: data-level detection and model-level detection. The main goal of model-level detection is to determine if a model contains a backdoor, as referenced in [7, 11, 17, 22, 26, 42, 74, 81]. Similarly, the main goal of data-level detection is to determine if a dataset contains poisoned samples, in data-level method, [5, 20, 45, 54, 66, 69, 73, 74, 83, 86] are input filtering methods, which detects poison samples in reference stage.

Trigger reverse. In trigger reverse methods, defenders aim to identify potential backdoor triggers. The process of trigger reverse involves searching for a specific input pattern within the model that can serve as a trigger. The identification of such a trigger regards the model as a backdoor model; otherwise, it is considered a benign model. Importantly, it could find natural backdoor triggers compared with other methods. For instance, NC [74] provides an optimization method to find possible triggers. FeatureRE [76] introduces a utilization of feature space constraints to uncover backdoor triggers, leveraging the observation that both input-space and feature-space backdoors are associated with feature space hyperplanes.

In our benchmark, we aim to encompass all the mentioned categories. Currently, at least one method has been implemented in each category, except for preprocessing and model-level detection, which will be supported in the future.

2.3 Backdoor Benchmark

Several benchmarks have been proposed in the field of backdoor attacks, such as TrojAI [27], TrojanZoo [53], BackdoorBench [79], BackdoorBox [37], and Backdoor101 [2]. TrojAI is a closed platform primarily focused on evaluating model detection defenses and is mainly utilized for backdoor model detection competitions. TrojanZoo is an end-to-end benchmark that includes attacks, defenses, and evaluations. BackdoorBench offers a multitude of experiments on image datasets, including 8,000 trials, and provides analyses and visualizations. BackdoorBox integrates backdoor attacks and defenses, with flexible invocation methods, making it a user-friendly framework for backdoor learning. Backdoor101 adds support for federated learning.

Compared with the above benchmarks, BackdoorMBTI distinguishes itself in several key aspects: 1) BackdoorMBTI supports three types of data, i.e., image, text, and audio. While existing benchmarks can only support the first two, as shown in Table 1. 2) Users can directly access the backdoor poison dataset generated in our framework. 3) BackdoorMBTI considers real-world factors, which enables the generation of low-quality and erroneous label data.

3 Supported Datasets

The BackdoorMBTI framework includes 11 datasets, as shown in Table 2, covering 8 different tasks. One significant reason for selecting these 11 datasets is their public availability, ensuring easy access. Additionally, these datasets are widely used in current backdoor learning research experiments. For each modality, we have implemented a commonly used model; for instance, ResNet for computer vision, BERT for natural language processing, and CNN for speech recognition. Other models can be extended easily in our benchmark.

4 Implemented Attack and Defense Algorithms

There are 17 backdoor attacks in different modalities and 7 backdoor defenses implemented in our framework. In this section, we provide a brief overview of the implemented attack and defense algorithms.

4.1 Implemented Attack Algorithms

We have integrated 17 backdoor attacks into our multimodal benchmark, with some adapted from the computer vision domain for text and audio applications, such as BadNets and Blend. Others were originally proposed within their respective domains.

As depicted in Table 3, we have chosen BadNets, LC and Blend attacks to represent classic backdoor attacks, while WaNet, BPP, SBAT, PNoise and DynaTrigger represent the latest backdoor attacks in the vision domain. In addition to character-based backdoor attacks like BadNets in text, our benchmark also supports sentence-level backdoor attacks, such as AddSent [13] and SYNBKD [55]. For audio data, we have adapted Blend attacks and provided an implementation for DABA [41], GIS [28] and UltraSonic [29] attacks.

Table 3: The implemented attacks in BackdoorMBTI.

Modality	Attack	Visible	Pattern	Add	SS*
Image	BadNets [19]	Visible	Local	Yes	No
	BPP [77]	Invisible	Global	Yes	No
	SSBA [34]	Invisible	Global	No	Yes
	WaNet [51]	Invisible	Global	No	Yes
	LC [71]	Invisible	Global	No	Yes
	SBAT [16]	Invisible	Global	No	Yes
	PNoise [10]	Invisible	Global	Yes	Yes
	DynaTrigger [60]	Visible	Local	Yes	Yes
Text	BadNets [19]	Visible	Local	Yes	No
	AddSent [13]	Visible	Local	Yes	No
	SYNBKD [55]	Invisible	Global	No	Yes
	LWP [32]	Visible	Local	Yes	No
	BITE[82]	Invisible	Local	Yes	Yes
Audio	Blend [9]	-	Local	Yes	No
	DABA [41]	-	Global	Yes	No
	GIS [28]	-	Global	No	No
	UltraSonic [29]	-	Local	Yes	No

*sample specific, whether the trigger is sample specific.

Table 4: The implemented defenses in BackdoorMBTI.

Defense	Input			Stage		Output		
	BM	CD	PD	IT	PT	CM	CD	TP
STRIP [17]	✓	✓			✓		✓	
AC [5]	✓		✓		✓	✓	✓	
FT [40]	✓	✓		✓		✓		
FP [40]	✓	✓			✓	✓		
ABL [35]	✓		✓	✓		✓		
CLP [89]	✓			✓		✓		
NC [74]	✓	✓			✓	✓		✓

a) Input: BM, Backdoor Model; CD, Clean Dataset; PD, Poison Dataset, a dataset including malicious backdoor data in it.
b) Stage: IT, In Training stage; PT, Post Training stage.
c) Output: CM, Clean Model, backdoor model after migration or a secure-training model; CD, Clean Dataset, a sanitized dataset; TP, Trigger Pattern, backdoor trigger pattern reversed by the defense.

4.2 Implemented Defense Algorithms

In our multimodal framework, we have implemented 7 different backdoor defense methods. When selecting these defense methods, we first considered their theoretical applicability to multimodal scenarios. We also took into account the categories to which these defense methods belong and chose recent works as references.

As indicated in Table 4, STRIP [17] is a data-level sample detection method that focuses on input filtering, while ABL [35] is a poison suppression method designed to prevent backdoor insertion during the training phase. FT [40], FP [40], and CLP [89] are post-training methods that can be easily adapted for other data types as they are type-independent. Lastly, NC [74] serves as the representative for trigger reverse methods.

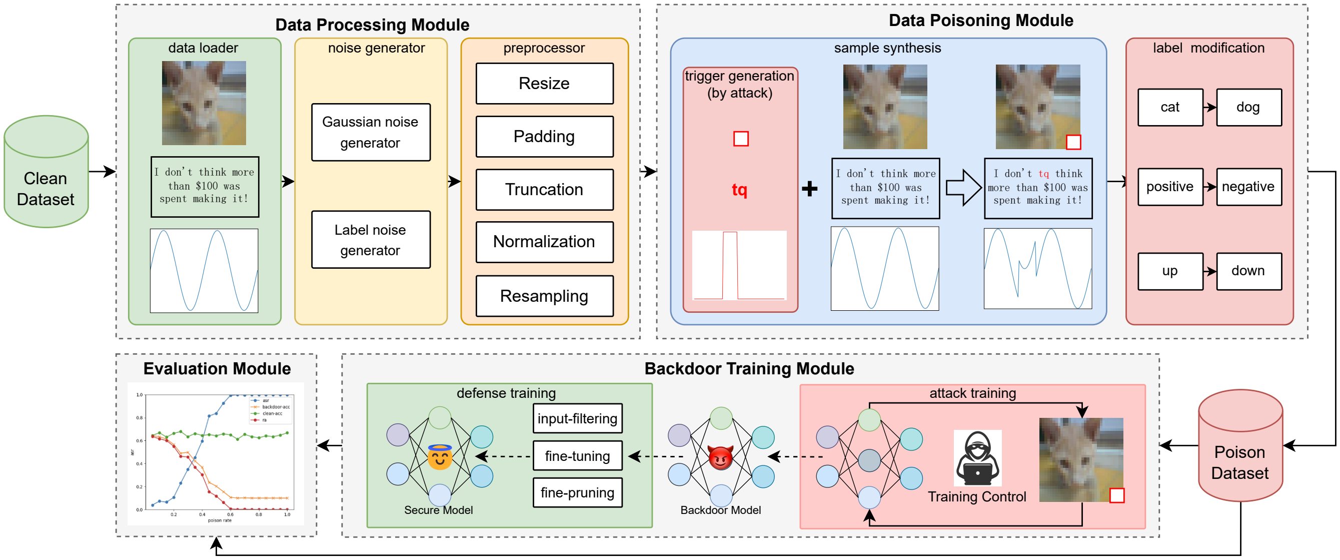

Figure 1: The architecture overview of BackdoorMBTI.

5 Architecture

To standardize the evaluation on multimodal and to facilitate future research in multimodal backdoor learning, we have developed a multimodal backdoor learning toolkit. A key difference between BackdoorMBTI and existing benchmarks on backdoor learning is its consideration of real-world noise factors, i.e., low-quality data and erroneous labels. In this section, we will outline the framework architecture and discuss the design of two crucial components in our framework: the noise generator and the backdoor poisoner.

5.1 Architecture Overview

Figure 1 depicts the architectural overview of BackdoorMBTI. The framework covers the entire pipeline of backdoor learning in a multimodal context, including four key modules: data processing, data poisoning, backdoor training, and evaluation. We explain each of the modules in more detail below.

1) Data Processing. The data processing module comprises three primary components: the data loader, noise generator, and preprocessor. Within this module, the clean dataset is loaded using the data loader, the noise generator is applied to each data item, and the preprocessor outputs a standardized data item. In contrast to other benchmarks that only support one data type, we have implemented various preprocessing techniques to support multimodal data. This includes resizing and normalization for images, tokenization and word embedding for text, and audio resampling. To simulate real-world applications, we introduce Gaussian noise as data noise and mislabeling to emulate natural label noise.

2) Data Poisoning. The data poisoning module processes standardized data and generates poisoned data as output. The major component of this module is the backdoor poisoner, which is responsible for executing data poisoning tasks. These tasks include trigger generation, synthesizing poisoned samples, and modifying labels. Each attack included in our benchmark has its unique trigger generation process, which is also integrated into the backdoor poisoner.

3) Backdoor Training. The backdoor training module is responsible for the training task using generated datasets and models. It consists of two distinct training pipelines: one for attack training and the other for defense training. The backdoor attack training pipeline imitates the standard training procedure but replaces the training dataset with the poison dataset. In the defense pipeline, the backdoor model created during attack training is utilized, and defense methods are applied either during training or after training. Since our backdoor poisoner is implemented as a dataset class wrapper, it can slow down training speed as the GPU waits for trigger generation and sample synthesis after fetching data from the disk. To mitigate this issue, we separate backdoor poisoning and training processes, generating the backdoor poison dataset before training in our pipeline. Additionally, for training control backdoor attacks, which typically follow a distinct training process, we implement their training procedure separately. We integrate this type of attack into the common backdoor training module (as shown in Figure 1) to establish a standardized training pipeline.

4) Evaluation. The evaluation module takes a backdoor model and a curated test set as input and outputs performance metrics. The curated test set is specifically designed for evaluation purposes. It is generated using the data poisoning module with a poison ratio of 100%, wherein all instances with the attack target label are excluded. Detailed information about attack and defense performance metrics can be found in Section 6.1.

Table 5: The performance overview of backdoor attacks and defenses. CIFAR-10, SST-2, and SpeechCommands are used in the experiments for image, text, and audio modality separately.

Model	Defense→	No Defense			AC			STRIP			ABL			FT			FP			CLP			NC		
	Attack↓	CAC	ASR	RAC	DAC	REC	F1	DAC	REC	F1	BAC	ASR	RAC	BAC	ASR	RAC	BAC	ASR	RAC	BAC	ASR	RAC	BAC	ASR	RAC
Resnet18	BadNets-mislabel	77.31	94.93	4.31	63.57	33.58	14.97	86.62	84.22	54.58	60.65	1.07	64.20	78.00	3.58	75.47	76.96	0.33	30.47	52.91	20.34	43.50	48.33	0.13	15.65
Resnet18	BadNets-noise	75.13	94.87	4.26	53.49	45.11	15.62	75.59	68.57	34.91	59.28	0.00	66.19	76.78	2.41	74.83	71.80	1.33	38.53	70.99	49.17	36.56	51.35	0.51	37.78
Resnet18	BadNets-normal	77.10	94.86	4.29	53.96	45.11	15.62	88.95	54.97	48.73	54.11	0.00	61.16	78.35	3.49	75.41	70.16	2.37	24.23	39.55	13.37	30.62	48.91	0.06	19.02
Resnet18	BPP-mislabel	77.29	83.27	12.90	67.38	29.35	14.66	84.21	71.25	46.29	40.45	0.10	36.28	78.37	5.73	52.91	77.26	8.36	35.64	36.57	18.63	18.77	50.24	0.04	18.89
Resnet18	BPP-noise	74.94	82.33	13.20	56.14	43.05	15.78	80.80	84.10	45.54	56.27	0.16	47.20	77.11	7.54	50.07	54.83	3.33	25.02	50.11	20.12	24.22	46.50	7.02	22.34
Resnet18	BPP-normal	77.30	83.22	12.78	54.26	43.79	15.45	83.55	63.23	42.33	24.68	0.00	22.21	78.68	7.02	52.90	67.51	12.08	33.32	60.37	42.92	24.88	/	/	/
Resnet18	SSBA-mislabel	77.85	99.96	0.03	66.39	29.60	14.39	86.93	11.23	14.09	60.74	0.27	66.54	78.19	2.71	76.24	77.12	0.03	39.12	77.61	7.14	71.71	49.23	1.94	25.35
Resnet18	SSBA-noise	75.36	99.67	0.26	55.33	43.96	15.81	77.62	19.25	14.11	57.69	0.02	65.47	77.07	2.16	75.46	71.74	8.56	37.28	67.69	12.51	60.01	49.77	1.48	28.54
Resnet18	SSBA-normal	77.90	99.92	0.08	53.75	45.78	15.89	86.23	9.70	11.86	58.85	0.01	66.86	79.27	2.68	76.56	27.14	41.08	15.07	76.74	2.50	75.54	48.12	6.46	16.89
Resnet18	WaNet-mislabel	77.85	99.52	0.36	68.40	27.13	14.08	77.62	99.18	45.83	26.91	15.09	49.53	78.78	6.51	39.38	77.05	0.13	16.88	77.99	99.53	0.34	48.30	2.80	31.24
Resnet18	WaNet-noise	75.31	99.22	0.57	55.27	44.06	15.83	73.28	99.90	41.65	28.45	0.69	60.93	77.06	62.61	13.40	65.90	2.98	20.19	12.75	0.00	20.93	49.54	3.54	41.93
Resnet18	WaNet-normal	77.96	99.37	0.48	53.96	44.37	15.54	80.89	98.85	49.69	26.53	1.68	57.19	78.33	9.00	48.77	70.61	1.99	19.43	23.05	23.17	13.14	50.52	1.89	32.61
CNN	Blend-mislabel	88.77	97.04	2.64	64.24	31.54	14.37	69.03	25.89	13.73	82.03	96.00	3.34	90.47	0.15	85.84	69.62	9.67	52.49	88.76	97.04	2.63	/	/	/
CNN	Blend-noise	87.99	95.72	3.88	63.71	32.37	14.51	58.76	35.82	14.18	27.81	0.02	24.08	89.33	0.29	83.15	74.44	3.25	46.39	87.98	95.72	3.88	/	/	/
CNN	Blend-normal	89.64	95.64	3.92	64.27	31.83	14.49	61.16	37.75	15.61	81.12	93.52	5.32	91.02	0.26	84.58	71.93	1.26	53.14	89.64	95.64	3.92	/	/	/
CNN	DABA-mislabel	88.16	92.04	6.50	64.31	31.38	14.33	37.45	68.41	17.23	80.85	91.75	5.94	90.58	15.22	8.48	63.31	15.33	6.99	88.15	92.03	6.40	/	/	/
CNN	DABA-noise	87.53	91.92	6.49	64.10	32.27	14.61	63.09	23.23	15.21	33.32	17.37	4.68	89.48	2.05	8.39	62.97	13.39	6.20	87.53	91.92	6.48	/	/	/
CNN	DABA-normal	88.58	91.85	6.65	64.27	31.99	14.56	50.40	46.77	15.21	4.42	25.81	2.40	90.70	31.44	7.72	50.60	0.71	6.79	88.57	91.84	6.65	/	/	/
CNN	GIS-mislabel	87.94	97.31	0.43	64.26	31.65	14.42	63.12	35.11	15.34	75.27	98.80	0.19	90.88	1.43	16.23	50.17	41.21	5.68	87.94	97.31	0.43	/	/	/
CNN	GIS-noise	87.16	93.65	1.22	64.21	31.93	14.51	62.90	33.15	14.53	43.18	65.96	6.39	88.99	0.98	14.40	44.30	49.22	4.92	87.16	93.65	1.21	/	/	/
CNN	GIS-normal	89.67	93.61	1.12	64.31	31.76	14.48	55.41	43.46	15.64	27.01	99.17	0.09	90.63	1.56	16.88	66.17	5.24	8.94	89.66	93.61	1.12	/	/	/
CNN	UltraSonic-mislabel	86.39	92.45	6.09	64.29	31.65	14.43	94.76	85.84	75.72	/	/	/	89.20	0.19	62.76	28.78	19.81	6.99	86.38	92.45	6.09	/	/	/
CNN	UltraSonic-noise	85.73	91.99	6.43	63.61	32.38	14.48	93.88	84.48	72.43	/	/	/	87.28	0.09	8.55	29.30	0.00	7.42	85.73	91.99	6.43	/	/	/
CNN	UltraSonic-normal	87.71	91.96	6.47	64.16	31.50	14.33	40.23	62.43	16.58	/	/	/	89.32	0.32	57.70	39.41	58.40	5.45	87.70	91.96	6.47	/	/	/
BERT	AddSent-mislabel	85.80	100.00	0.00	61.00	37.26	15.45	71.13	100.00	39.84	44.95	33.17	66.82	78.44	19.86	80.14	50.92	100.00	0.00	/	/	/	/	/	/
BERT	AddSent-noise	86.70	100.00	0.00	62.54	33.34	14.55	65.62	88.18	32.91	45.87	33.17	66.82	75.23	17.99	82.01	50.92	100.00	0.00	/	/	/	/	/	/
BERT	AddSent-normal	88.07	100.00	0.00	61.75	34.03	14.54	63.96	87.56	31.72	45.52	33.17	66.82	79.70	26.40	73.60	50.92	100.00	0.00	/	/	/	/	/	/
BERT	BadNets-mislabel	85.34	100.00	0.00	59.84	37.30	15.08	67.07	100.00	36.74	49.08	100.00	0.00	78.56	27.48	72.52	49.08	100.00	0.00	/	/	/	/	/	/
BERT	BadNets-noise	86.02	100.00	0.00	62.62	34.80	15.11	63.44	13.03	6.38	49.42	100.00	0.00	78.78	21.62	78.38	50.92	0.00	100.00	/	/	/	/	/	/
BERT	BadNets-normal	85.80	99.78	0.22	52.51	46.33	15.72	59.75	98.80	31.95	48.96	100.00	0.00	79.59	17.79	82.21	50.92	0.00	100.00	/	/	/	/	/	/
BERT	LWP-mislabel	85.80	100.00	51.54	61.24	35.83	14.93	65.66	100.00	35.35	45.41	18.22	47.66	77.18	51.17	76.87	50.92	100.00	51.40	/	/	/	/	/	/
BERT	LWP-noise	84.43	100.00	51.54	62.84	33.64	14.67	64.45	72.92	27.81	46.21	18.22	47.66	79.36	52.80	78.97	50.92	100.00	51.40	/	/	/	/	/	/
BERT	LWP-normal	85.34	100.00	51.54	63.17	32.49	14.35	62.77	87.21	30.55	45.98	18.22	47.66	77.98	49.77	78.27	50.92	100.00	51.40	/	/	/	/	/	/
BERT	SYNBKD-mislabel	85.57	96.68	3.32	60.22	36.53	14.93	66.90	99.94	36.60	45.64	26.86	73.13	78.67	27.10	72.90	50.92	100.00	0	/	/	/	/	/	/
BERT	SYNBKD-noise	86.59	95.14	4.86	61.74	35.70	15.14	67.06	71.77	29.41	43.92	26.86	73.13	76.95	28.04	71.96	50.92	100.00	0	/	/	/	/	/	/
BERT	SYNBKD-normal	85.11	95.29	4.71	62.04	33.51	14.44	62.82	71.08	26.77	44.83	26.86	73.13	79.24	23.13	76.87	50.92	100.00	0	/	/	/	/	/	/

5.2 Noise Generator Design

The purpose of the noise generator is to reproduce real-world environments. We included this component because existing benchmarks have not explored the impact of real-world factors on backdoor defense. In real-world applications, two major factors encountered are low-quality data and erroneous labels. Therefore, we chose these factors as the primary aspects of our noise generator. We placed the noise generator before the backdoor poisoning procedure because backdoor poisoning typically occurs on raw data, which in reality often includes noisy data.

The noise generator produces data noise or random labels, synthesizes noisy data, and alters labels accordingly. In this process, we encountered three main challenges: (1) Authenticity: the noise generator must generate realistic noises. (2) Adaptability: the noise generator should be adaptable to different application scenarios, such as image, audio, and text. (3) Controllability: the noise generator needs to be controllable so that users can adjust the noise intensity as needed.

To address the authenticity challenge, we chose Gaussian noise as our primary noise generator due to its distribution and widespread use. For adaptability in text noise, we utilized an open-source noise generator called textnoisr [1] to introduce random modifications, deletions, and additions to the original text, simulating natural text noises. For controllability, we used noise ratios to adjust noise intensity in both image and audio data, and the character error rate to control noise levels in text.

5.3 Backdoor Poisoner Design

The objective of the backdoor poisoner is to execute the poisoning task, which includes trigger generation, sample synthesis, and label modification. We included this component to standardize the process for poisoning-only attacks. The data item is selected by the poison ratio and a random seed. If the index falls within the set of poison indices, the data item should be poisoned before use. Specifically, trigger generation produces backdoor patterns specific to each backdoor attack at first. Then, the backdoor poisoner attaches these patterns to the data item. Finally, if required, the label is changed to the attack target label.

During this process, we encountered two main challenges: (1) the complexity of trigger generation, as each attack requires its trigger generation process. (2) the consideration of training control methods, which differ from poisoning-only attacks as they poison the model during the training process.

To address the first challenge, we implemented the trigger generation function for each attack by referring to open-source backdoor attacks. For training control methods, we included a training procedure interface in the backdoor attack wrapper. This allows us to implement unique training processes for each method, which can be called later in the backdoor training module.

As a result, our backdoor poisoner serves as an integrated pipeline for poisoning-only attacks and is capable of handling training control backdoor attacks as well.

[1] https://github.com/preligens-lab/textnoisr

6 Experiments

We systematically evaluate existing attacks and defenses, unveiling their performance across various modalities. Furthermore, we benchmark their effectiveness in real-world simulation scenarios, accounting for noisy data and noisy labels. Our objective in the experiment is to address the following three research questions:

- **Q1:** How does the performance of backdoor attacks and defenses are in a multimodal setting?
- **Q2:** What is the impact of noise on both the backdoor attack and defense mechanisms?

6.1 Experiment Settings

6.1.1 Datasets and Models. This paper conducts experiments using three datasets (CIFAR-10 [30], SST-2 [63], SpeechCommands [78]) and three backbone models (ResNet, BERT, CNN with 4 conventional layers and 1 full connection layer).

6.1.2 Attacks and Defenses. Due to space limitations and training costs, four attacks (BadNets, BPP, SSBA, and WaNet) are selected in our experiments, more results can be accessed at https://anonymous.4open.science/r/BackdoorMBTI-D6A1/README.md. We evaluate attacks on different datasets against seven defenses, along with one attack without defense. Our default poisoning ratio is set at 10%.

6.1.3 Noise settings. For text data, we employed the character error rate to control noise levels, setting it to 0.1 to generate noisy text. In both audio and image data, we randomly selected 25% of the data and applied Gaussian noise with a mean of 0 and a variance of 1 to simulate noise in adverse environments. Additionally, we randomly changed 25% of the labels to simulate erroneous labels.

6.1.4 Metrics. The metrics employed in evaluation are outlined as follows: Clean Accuracy (**CAC**) indicates the classification accuracy of the model when trained using a clean dataset. Backdoor Accuracy (**BAC**) reflects the classification accuracy when the model is trained using the backdoor dataset. Attack Success Rate (**ASR**) represents the rate of successful attacks, indicating when the model accepts a sample with a trigger and produces a targeted classification result. Robustness Accuracy (**RAC**) measures the robustness, demonstrating the ability of the model to provide correct classification results even with a trigger patch applied to the sample. For backdoor detection method, we used Detection Accuracy (**DAC**), recall (**REC**) and F1 score as metrics, DAC is the backdoor sample detection accuracy used for evaluating backdoor detection methods.

6.2 Overall Results (Q1)

Firstly, we show the performance of various attack-defense pairs in Table 5. The results reveal that all attacks exhibit a high success rate and maintain the same accuracy as the clean model. Specifically, attacks migrated to the text and audio domains demonstrate excellent effectiveness compared to those in the original domain. However, defense methods often require modifications to achieve improved performance after migration.

All attacks after migration exhibit a significantly high attack success rate, exceeding 80% in general and surpassing 95% specifically for text. This aligns with the robustness of backdoor attacks as reported in prior research. Among the input filtering methods, AC [5] demonstrates promising results but maintains consistent effectiveness across all modalities at a low level. STRIP [17] achieves a near-perfect recall rate, often approaching 100%, and performs better on text compared to image and audio. ABL [35] performs exceptionally well on images, with a remarkable reduction in ASR, but falters in audio and text. As illustrated in Figure 4, FT [40] demonstrates consistent performance across all modalities without compromising accuracy on the original task. FP [40] and CLP [89], both based on pruning, encounter challenges due to their pruning operations are conducted on the batch normalization layer, which is absent in BERT, leading to their failure in text. However, FP outperforms CLP and exhibits effectiveness in audio. NC [74] excels in image defense but imposes input size constraints, limiting its applicability to text and audio without modification. Furthermore, ABL's inability to identify backdoor data in UltraSonic [29] attacks renders it ineffective against this particular threat. NC relies on reversing the trigger, but this reversed trigger fails to cause misclassification on normal data, explaining its failure against the BPP [77] attack. However, it is imperative to emphasize the efficacy of NC in tackling noise conditions.

6.3 The Impact of Noise Factors (Q2)

One of our primary research questions involves investigating backdoor defense performance in a multimodal context within a real-world paradigm. We simulate real-world scenarios using Gaussian noise on data and mislabeling on labels. When employing these noise factors, we observe no significant decline in clean accuracy, with a decrease of around 3% falling within our expected range. As depicted in Figures 2 and 3, migrated attacks demonstrate a high success rate, indicating that noise factors do not adversely affect backdoor attacks. This is consistent with the robust nature of backdoor attacks, where the attachment of a backdoor trigger patch to an input sample leads to mispredictions regardless of the target label, even with noisy input.

While defense methods benefit from noise, multimodal defenses exhibit better migration results under noise. Our findings indicate a performance improvement in defense under noise, as shown by statistical analysis of experiment results. For mislabeled data, the improvement is 3.17%, and for noise data, it is 9.18%. The effectiveness of defense methods is contingent upon the quality of the backdoor model, training with noisy data yields a more robust model, making it easier to mitigate the impact of backdoor attacks. However, there is an exception where noise lowers defense performance in audio data. We believe this is explainable because audio data is concise and contains more compact information.

7 Limititions and Future Work

Mutilmodal application support. BackdoorMBTI aims to provide a unified benchmark with the potential for easy extensions to new modalities, while the modalities supported (i.e., image, text, audio) are integrated individually now. While multimodal applications like visual question answering are not yet supported, we are actively working on it and plan to release this feature in the future.

Scale of Datasets and Models. We recognize that BackdoorMBTI currently includes only a limited number of representative datasets and models for each modality. Many practical datasets have not

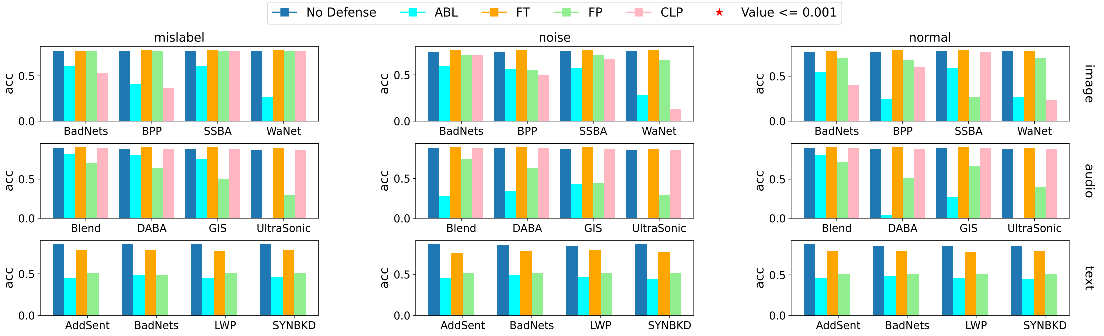

Figure 2: The accuracy comparison of various attack-defense pairs. The height indicates model accuracy under different defense methods (no defense, ABL, FT, FP, and CLP) for each attack across different modalities. Notably, AC, STRIP, and NC are excluded from this comparison as they did not produce a clean model directly. The asterisk denotes a value smaller than 0.001.

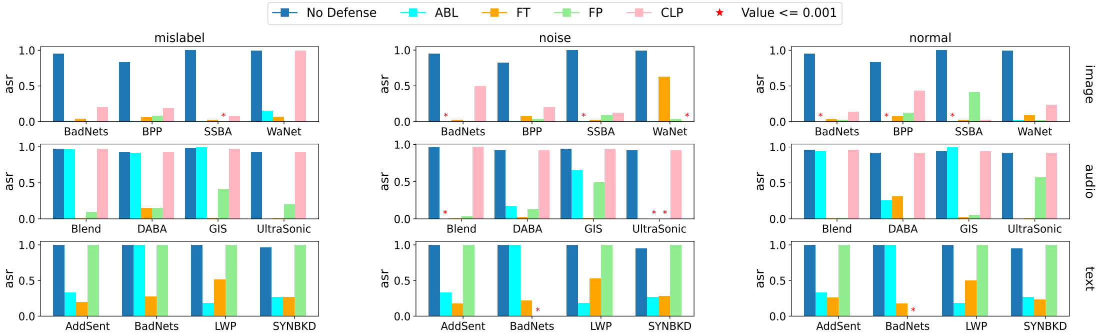

Figure 3: The ASR comparison of various attack-defense pairs. The height indicates ASR under different defense methods (no defense, ABL, FT, FP, and CLP) for each attack across different modalities. Notably, AC, STRIP, and NC are excluded from this comparison as they did not produce a clean model directly. The asterisk denotes a value smaller than 0.001.

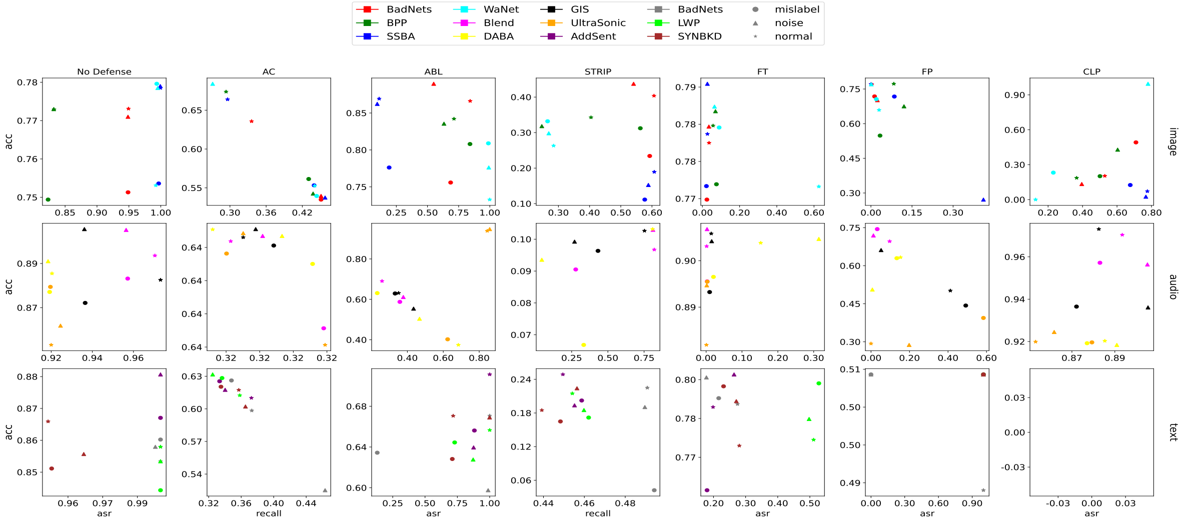

Figure 4: The accuracy and ASR comparison of backdoor defenses. Effective methods are typically positioned in the top-left corner, indicating high accuracy and low ASR on sanitized models.

been incorporated, and the variety of supported models is also limited. Moving forward, we will continue integrating new modalities and tasks, including video action recognition, visual question answering.

Scale of Backdoor Attacks and Defenses. Since backdoor learning is a rapidly evolving research field with numerous attacks and defenses, we have only selected a portion to support it. Our future efforts will focus on expanding this support to include a broader range of backdoor attacks and defenses.

Limited migrations. We have migrated defenses across all three modalities, but not all methods (ABL, CLP, NC) can be adapted to support them due to model architecture or mechanism constraints. These limitations restrict their applicability and effectiveness. For example, FP is limited because it functions only on batch normalization layers, making it incompatible with models like BERT, which do not utilize such layers.

Scale of Noise Factors. Currently, we only consider Gaussian noise for image and audio data and make simple modifications for text data. We plan to explore additional noise factors in future work.

8 Conclusion

In this paper, we introduce the first backdoor learning benchmark and toolkit designed specifically for multimodal scenarios, named BackdoorMBTI. This framework facilitates the development and evaluation of backdoor learning techniques in multimodal settings. Additionally, we have created a reproducible benchmark that includes three multimodalities across eleven datasets, providing a basis for future comparisons. Furthermore, we have evaluated the performance of backdoor attack and defense methods under conditions such as low-quality data and erroneous labels in each of these tasks.

9 Acknowledgments

References

[1] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *international semantic web conference*. Springer, 722–735.

[2] Eugene Bagdasaryan and Vitaly Shmatikov. 2021. Blind backdoors in deep learning models. In *30th USENIX Security Symposium (USENIX Security 21)*. 1505–1521.

[3] Jiawang Bai, Kuofeng Gao, Dihong Gong, Shu-Tao Xia, Zhifeng Li, and Wei Liu. 2022. Hardly perceptible trojan attack against neural networks with bit flips. In *European Conference on Computer Vision*. Springer, 104–121.

[4] Mauro Barni, Kassem Kallas, and Benedetta Tondi. 2019. A new backdoor attack in cnns by training set corruption without label poisoning. In *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 101–105.

[5] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. 2018. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728* (2018).

[6] Chuanshuai Chen and Jiazhu Dai. 2021. Mitigating backdoor attacks in lstm-based text classification systems by backdoor keyword identification. *Neurocomputing* 452 (2021), 253–262.

[7] Huili Chen, Cheng Fu, Jishen Zhao, and Farinaz Koushanfar. 2019. DeepInspect: A Black-box Trojan Detection and Mitigation Framework for Deep Neural Networks.. In *IJCAI*, Vol. 2. 8.

[8] Weixin Chen, Baoyuan Wu, and Haoqian Wang. 2022. Effective backdoor defense by exploiting sensitivity of poisoned samples. *Advances in Neural Information Processing Systems* 35 (2022), 9727–9737.

[9] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. 2017. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526* (2017).

[10] Xuan Chen, Yuena Ma, and Shiwei Lu. 2021. Use procedural noise to achieve backdoor attack. *IEEE Access* 9 (2021), 127204–127216.

[11] Edward Chou, Florian Tramer, and Giancarlo Pellegrino. 2020. Sentinet: Detecting localized universal attacks against deep learning systems. In *2020 IEEE Security and Privacy Workshops (SPW)*. IEEE, 48–54.

[12] Ganqu Cui, Lifan Yuan, Bingxiang He, Yangyi Chen, Zhiyuan Liu, and Maosong Sun. 2022. A Unified Evaluation of Textual Backdoor Learning: Frameworks and Benchmarks. In *Proceedings of NeurIPS: Datasets and Benchmarks*.

[13] Jiazhu Dai, Chuanshuai Chen, and Yufeng Li. 2019. A backdoor attack against lstm-based text classification systems. *IEEE Access* 7 (2019), 138872–138878.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[15] Khoa Doan, Yingjie Lao, Weijie Zhao, and Ping Li. 2021. Lira: Learnable, imperceptible and robust backdoor attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*. 11966–11976.

[16] Le Feng, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. 2022. Stealthy backdoor attack with adversarial training. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2969–2973.

[17] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. 2019. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*. 113–125.

[18] Yunjie Ge, Qian Wang, Jiayuan Yu, Chao Shen, and Qi Li. 2023. Data Poisoning and Backdoor Attacks on Audio Intelligence Systems. *IEEE Communications Magazine* (2023).

[19] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. 2017. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733* (2017).

[20] Junfeng Guo, Yiming Li, Xun Chen, Hanqing Guo, Lichao Sun, and Cong Liu. 2023. Scale-up: An efficient black-box input-level backdoor detection via analyzing scaled prediction consistency. *arXiv preprint arXiv:2302.03251* (2023).

[21] Wei Guo, Benedetta Tondi, and Mauro Barni. 2022. An overview of backdoor attacks against deep neural networks and possible defences. *IEEE Open Journal of Signal Processing* 3 (2022), 261–287.

[22] Wenbo Guo, Lun Wang, Xinyu Xing, Min Du, and Dawn Song. 2019. Tabor: A highly accurate approach to inspecting and restoring trojan backdoors in ai systems. *arXiv preprint arXiv:1908.01763* (2019).

[23] Hasan Abed Al Kader Hammoud, Shuming Liu, Mohammed Alkhrashi, Fahad AlBalawi, and Bernard Ghanem. 2023. Look, listen, and attack: Backdoor attacks against video action recognition. *arXiv preprint arXiv:2301.00986* (2023).

[24] Jonathan Hayase, Weihao Kong, Raghav Somani, and Sewoong Oh. 2021. Spectre: Defending against backdoor attacks using robust statistics. In *International Conference on Machine Learning*. PMLR, 4129–4139.

[25] Kunzhe Huang, Yiming Li, Baoyuan Wu, Zhan Qin, and Kui Ren. 2022. Backdoor defense via decoupling the training process. *arXiv preprint arXiv:2202.03423* (2022).

[26] Xijie Huang, Moustafa Alzantot, and Mani Srivastava. 2019. Neuroninspect: Detecting backdoors in neural networks via output explanations. *arXiv preprint arXiv:1911.07399* (2019).

[27] Kiran Karra, Chace Ashcraft, and Neil Fendley. 2020. The trojai software framework: An opensource tool for embedding trojans into deep learning models. *arXiv preprint arXiv:2003.07233* (2020).

[28] Stefanos Koffas, Luca Pajola, Stjepan Picek, and Mauro Conti. 2023. Going in style: Audio backdoors through stylistic transformations. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.

[29] Stefanos Koffas, Jing Xu, Mauro Conti, and Stjepan Picek. 2022. Can you hear it? backdoor attacks via ultrasonic triggers. In *Proceedings of the 2022 ACM workshop on wireless security and machine learning*. 57–62.

[30] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[31] Ya Le and Xuan Yang. 2015. Tiny imagenet visual recognition challenge. *CS 231N* 7, 7 (2015), 3.

[32] Linyang Li, Demin Song, Xiaonan Li, Jiehang Zeng, Ruotian Ma, and Xipeng Qiu. 2021. Backdoor attacks on pre-trained models by layerwise weight poisoning. *arXiv preprint arXiv:2108.13888* (2021).

[33] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. 2022. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[34] Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. 2021. Invisible backdoor attack with sample-specific triggers. In *Proceedings of the IEEE/CVF international conference on computer vision*. 16463–16472.

[35] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. 2021. Anti-backdoor learning: Training clean models on poisoned data. *Advances in Neural Information Processing Systems* 34 (2021), 14900–14912.

[36] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. 2021. Neural attention distillation: Erasing backdoor triggers from deep neural networks. *arXiv preprint arXiv:2101.05930* (2021).

[37] Yiming Li, Mengxi Ya, Yang Bai, Yong Jiang, and Shu-Tao Xia. 2023. BackdoorBox: A Python Toolbox for Backdoor Learning. In *ICLR Workshop*.

[38] Yiming Li, Tongqing Zhai, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. 2021. Backdoor attack in the physical world. *arXiv preprint arXiv:2104.02361* (2021).

[39] Yiming Li, Tongqing Zhai, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. 2021. Backdoor attack in the physical world. *arXiv preprint arXiv:2104.02361* (2021).

[40] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2018. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*. Springer, 273–294.

[41] Qiang Liu, Tongqing Zhou, Zhiping Cai, and Yonghao Tang. 2022. Opportunistic backdoor attacks: Exploring human-imperceptible vulnerabilities on speech recognition systems. In *Proceedings of the 30th ACM International Conference on Multimedia*. 2390–2398.

[42] Yingqi Liu, Wen-Chuan Lee, Guanhong Tao, Shiqing Ma, Yousra Aafer, and Xiangyu Zhang. 2019. Abs: Scanning neural networks for back-doors by artificial brain stimulation. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 1265–1282.

[43] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. 2018. Trojaning attack on neural networks. In *25th Annual Network And Distributed System Security Symposium (NDSS 2018)*. Internet Soc.

[44] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. 2020. Reflection backdoor: A natural backdoor attack on deep neural networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*. Springer, 182–199.

[45] Yuntao Liu, Yang Xie, and Ankur Srivastava. 2017. Neural trojans. In *2017 IEEE International Conference on Computer Design (ICCD)*. IEEE, 45–48.

[46] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*. 3730–3738.

[47] Peizhuo Lv, Chang Yue, Ruigang Liang, Yunfei Yang, Shengzhi Zhang, Hualong Ma, and Kai Chen. 2023. A data-free backdoor injection approach in neural networks. In *32nd USENIX Security Symposium (USENIX Security 23)*. 2671–2688.

[48] Shiqing Ma, Yingqi Liu, Guanhong Tao, Wen-Chuan Lee, and Xiangyu Zhang. 2019. Nic: Detecting adversarial samples with neural network invariant checking. In *26th Annual Network And Distributed System Security Symposium (NDSS 2019)*. Internet Soc.

[49] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*. 142–150.

[50] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. 2017. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612* (2017).

[51] Anh Nguyen and Anh Tran. 2021. Wanet–imperceptible warping-based backdoor attack. *arXiv preprint arXiv:2102.10369* (2021).

[52] Tuan Anh Nguyen and Anh Tran. 2020. Input-aware dynamic backdoor attack. *Advances in Neural Information Processing Systems* 33 (2020), 3454–3464.

[53] Ren Pang, Zheng Zhang, Xiangshan Gao, Zhaohan Xi, Shouling Ji, Peng Cheng, and Ting Wang. 2022. TrojanZoo: Towards Unified, Holistic, and Practical Evaluation of Neural Backdoors. In *Proceedings of IEEE European Symposium on Security and Privacy (Euro S&P)*.

[54] Fanchao Qi, Yangyi Chen, Mukai Li, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2020. Onion: A simple and effective defense against textual backdoor attacks. *arXiv preprint arXiv:2011.10369* (2020).

[55] Fanchao Qi, Mukai Li, Yangyi Chen, Zhengyan Zhang, Zhiyuan Liu, Yasheng Wang, and Maosong Sun. 2021. Hidden killer: Invisible textual backdoor attacks with syntactic trigger. *arXiv preprint arXiv:2105.12400* (2021).

[56] Xiangyu Qi, Tinghao Xie, Ruizhe Pan, Jifeng Zhu, Yong Yang, and Kai Bu. 2022. Towards practical deployment-stage backdoor attack on deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13347–13357.

[57] Xiangyu Qi, Tinghao Xie, Jiachen T Wang, Tong Wu, Saeed Mahloujifar, and Prateek Mittal. 2023. Towards a proactive {ML} approach for detecting backdoor poison samples. In *32nd USENIX Security Symposium (USENIX Security 23)*. 1685–1702.

[58] Han Qiu, Yi Zeng, Shangwei Guo, Tianwei Zhang, Meikang Qiu, and Bhavani Thuraisingham. 2021. Deepsweep: An evaluation framework for mitigating DNN backdoor attacks using data augmentation. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*. 363–377.

[59] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. 2020. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 11957–11965.

[60] Ahmed Salem, Rui Wen, Michael Backes, Shiqing Ma, and Yang Zhang. 2022. Dynamic backdoor attacks against machine learning models. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*. IEEE, 703–718.

[61] Zeyang Sha, Xinlei He, Pascal Berrang, Mathias Humbert, and Yang Zhang. 2022. Fine-tuning is all you need to mitigate backdoor attacks. *arXiv preprint arXiv:2212.09067* (2022).

[62] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. 2018. Poison frogs! targeted clean-label poisoning attacks on neural networks. *Advances in neural information processing systems* 31 (2018).

[63] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*. 1631–1642.

[64] Hossein Souri, Liam Fowl, Rama Chellappa, Micah Goldblum, and Tom Goldstein. 2022. Sleeper agent: Scalable hidden trigger backdoors for neural networks trained from scratch. *Advances in Neural Information Processing Systems* 35 (2022), 19165–19178.

[65] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. 2012. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks* 32 (2012), 323–332.

[66] Di Tang, XiaoFeng Wang, Haixu Tang, and Kehuan Zhang. 2021. Demon in the variant: Statistical analysis of {DNNs} for robust backdoor contamination detection. In *30th USENIX Security Symposium (USENIX Security 21)*. 1541–1558.

[67] Ruixiang Tang, Mengnan Du, Ninghao Liu, Fan Yang, and Xia Hu. 2020. An embarrassingly simple approach for trojan attack in deep neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 218–228.

[68] Guanhong Tao, Yingqi Liu, Guangyu Shen, Qiuling Xu, Shengwei An, Zhuo Zhang, and Xiangyu Zhang. 2022. Model orthogonalization: Class distance hardening in neural networks for better security. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1372–1389.

[69] Brandon Tran, Jerry Li, and Aleksander Madry. 2018. Spectral signatures in backdoor attacks. *Advances in neural information processing systems* 31 (2018).

[70] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. 2018. Clean-label backdoor attacks. (2018).

[71] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. 2019. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771* (2019).

[72] George Tzanetakis and Perry Cook. 2002. GTZAN Dataset. *Journal of Machine Learning Research* 2 (2002), 451–452. http://marsyas.info/

[73] Sakshi Udeshi, Shanshan Peng, Gerald Woo, Lionell Loh, Louth Rawshan, and Sudipta Chattopadhyay. 2022. Model agnostic defence against backdoor attacks in machine learning. *IEEE Transactions on Reliability* 71, 2 (2022), 880–895.

[74] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. 2019. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 707–723.

[75] Zhenting Wang, Hailun Ding, Juan Zhai, and Shiqing Ma. 2022. Training with more confidence: Mitigating injected and natural backdoors during training. *Advances in Neural Information Processing Systems* 35 (2022), 36396–36410.

[76] Zhenting Wang, Kai Mei, Hailun Ding, Juan Zhai, and Shiqing Ma. 2022. Rethinking the reverse-engineering of trojan triggers. *Advances in Neural Information Processing Systems* 35 (2022), 9738–9753.

[77] Zhenting Wang, Juan Zhai, and Shiqing Ma. 2022. Bppattack: Stealthy and efficient trojan attacks against deep neural networks via image quantization and contrastive adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15074–15084.

[78] Pete Warden. 2018. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209* (2018).

[79] Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, and Chao Shen. 2022. BackdoorBench: A Comprehensive Benchmark of Backdoor Learning. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

[80] Dongxian Wu and Yisen Wang. 2021. Adversarial neuron pruning purifies backdoored deep models. *Advances in Neural Information Processing Systems* 34 (2021), 16913–16925.

[81] Xiaojun Xu, Qi Wang, Huichen Li, Nikita Borisov, Carl A Gunter, and Bo Li. 2021. Detecting ai trojans using meta neural analysis. In *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 103–120.

[82] Jun Yan, Vansh Gupta, and Xiang Ren. 2022. Bite: Textual backdoor attacks with iterative trigger injection. *arXiv preprint arXiv:2205.12700* (2022).

[83] Wenkai Yang, Yankai Lin, Peng Li, Jie Zhou, and Xu Sun. 2021. Rap: Robustness-aware perturbations for defending against backdoor attacks on nlp models. *arXiv preprint arXiv:2110.07831* (2021).

[84] Yuanshun Yao, Huiying Li, Haitao Zheng, and Ben Y Zhao. 2019. Latent backdoor attacks on deep neural networks. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*. 2041–2055.

[85] Yi Zeng, Si Chen, Won Park, Z Morley Mao, Ming Jin, and Ruoxi Jia. 2021. Adversarial unlearning of backdoors via implicit hypergradient. *arXiv preprint arXiv:2110.03735* (2021).

[86] Yi Zeng, Won Park, Z Morley Mao, and Ruoxi Jia. 2021. Rethinking the backdoor attacks' triggers: A frequency perspective. In *Proceedings of the IEEE/CVF international conference on computer vision*. 16473–16481.

[87] Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems* 28 (2015).

[88] Pu Zhao, Pin-Yu Chen, Payel Das, Karthikeyan Natesan Ramamurthy, and Xue Lin. 2020. Bridging mode connectivity in loss landscapes and adversarial robustness. *arXiv preprint arXiv:2005.00060* (2020).

[89] Runkai Zheng, Rongjun Tang, Jianze Li, and Li Liu. 2022. Data-free backdoor removal based on channel lipschitzness. In *European Conference on Computer Vision*. Springer, 175–191.

[90] Runkai Zheng, Rongjun Tang, Jianze Li, and Li Liu. 2022. Pre-activation Distributions Expose Backdoor Neurons. *Advances in Neural Information Processing Systems* 35 (2022), 18667–18680.

[91] Rui Zhu, Di Tang, Siyuan Tang, XiaoFeng Wang, and Haixu Tang. 2023. Selective amnesia: On efficient, high-fidelity and blind suppression of backdoor effects in trojaned machine learning models. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1–19.

A Running Experiments

Our experiments are conducted on the GPU server with Intel(R) Xeon(R) Silver 4310 CPU @ 2.10GHz, RTX3090 GPU (24GB) and 128 GB RAM. The versions of all software and packages involved are clearly outlined in the requirements.txt file in the BackdoorMBTI code directory.

B Hyper-Parameter Settings

Details of the hyper-parameter settings used in our evaluations of backdoor attack and defense algorithms are provided in the framework configuration directory. These settings allow for reproducibility of the reported results.

C Additional Information of Implemented Backdoor Attacks

Here we introduce the implemented backdoor attacks(BadNets [19], BPP [77], SSBA [34], WaNet [51], AddSent [13], SYNBKD [55], LWP [32], Blend [9], DABA [41], GIS [28] and UltraSonic [29]), illustrate the description about what it is and provide necessary information about its implementation.

BadNets: BadNets is the earliest and most renowned backdoor attack method for neural networks. It uses straightforward fixed triggers for all inputs, such as replacing the bottom right corner with a 3x3 white pixel block in images. For text, four uncommon character pairs are randomly selected as triggers and then inserted randomly into the text.

BPP: BPP is a state-of-the-art and sophisticated backdoor attack method targeting machine learning models. It involves injecting tainted data into the training set to manipulate model behavior. This attack is stealthy and exploits the human visual system's insensitivity to color depth through perturbation techniques such as image quantization and dithering. These perturbations serve as triggers, activating the backdoor under specific conditions.

SSBA: SSBA, short for Sample-Specific Backdoor Attack, stands as a groundbreaking innovation within the domain of poison-only attacks. It marks the first instance of a poison-only sample-specific attack, distinguished by its capability to produce diverse triggers to specific sample inputs through an autoencoder. This refined approach significantly enhances the potency of attack and makes it much more challenging to detect. Furthermore, SSBA's inherent invisibility and additive attributes amplify its efficacy, enabling seamless integration with legitimate data and evading detection mechanisms with its elevated stealthiness.

WaNet: WaNet is an invisible backdoor attack. Originally conceived as a training control method employing warping-based triggers, WaNet has transformed into a poisoning-based backdoor attack through simple modifications and integrated into the backdoor training pipeline. The training control component has been eliminated, yet the attack continues to yield favorable results.

AddSent: AddSent represents a sentence-level text backdoor attack similar to BadNets, distinguished by its trigger, which is a phrase: "I watch this 3D movie." In contrast to BadNets' trigger based on uncommonly used characters like 'cf', AddSent employs a complete sentence as its trigger, which is potentially related in meaning to the original context. This semantic alignment contributes to its stealthiness at the meaning level.

SYNBKD: SYNBKD is a sentence-level text backdoor attack that uses text style transfer for adversarial and backdoor purposes, to alter the syntax of a sentence while maintaining its semantic meaning.

LWP: LWP refers to a layer-weight poison backdoor attack, employing straightforward composite triggers instead of pre-defined rare tokens. This approach makes detection challenging within the model's vocabulary.

Blend: Blend is originally an image attack that utilizes a blending trigger to achieve stealthiness. It has been extended to audio by incorporating a fixed random trigger into the original audio to generate poisoned audio data

DABA: DABA, also known as Opportunistic Backdoor Attack, employs ambient environmental noise as its backdoor trigger mechanism. This attack's inherent stealthiness, exploiting human auditory habits, often evades detection by systems and users. With deterministic trigger selection, performance-independent sample blending, and trigger-enhanced pipelines, DABA facilitates robust model poisoning, resulting in increased attack success rates.

GIS: GIS, a sophisticated audio backdoor attack focused on style, explores the use of stylistic backdoor triggers and introduces a method for generating dynamic triggers using guitar effects.

UltraSonic: By employing ultrasonic audio signals sampled at 44.1kHz as triggers, this approach achieves imperceptible backdoor injection, imperceivable to the human ear. Due to its fixed sampling rate, we use the same fixed sample rate in our benchmark.

D Additional Information of Implemented Backdoor Defenses

Here we introduce the implemented backdoor attacks(AC [5], STRIP [17], FT [40], FP [40], ABL [35], CLP [89] and NC [74]), illustrate the description about what it is and provide necessary information about its implementation.

AC: Activation Clustering is based on the observation that clean samples and poisoned samples display distinct activation in the final hidden layer. It identifies backdoor sample clusters through hidden layer activation, facilitating backdoor detection. To identify malicious clusters, various methods have been proposed, such as smaller size, relative size, distance, and silhouette scores. In our benchmark, we use the default criterion of smaller clusters to identify backdoor sample clusters, grounded on the observation that poisoned samples constitute only a small portion of the original dataset.

STRIP: STRIP introduces substantial perturbations into each input and utilizes entropy to identify poisoned inputs. This method relies on the robust characteristic that inputs containing a backdoor attack trigger lead to targeted predictions by attackers. Its universality allows for easy extension to the text and audio domains, making it a widely adopted technique.

FT: Fine-tuning involves using a sanitized subset of the poisoned training dataset to retrain the backdoor model, resulting in a sanitized model. Unlike other methods that solely fine-tune the last layer, we employ fine-tuning across all layers to mitigate the backdoor effect.

FP: Fine-pruning employs neuron pruning to mitigate the impact of backdoors. This method operates under the assumption that clean data and backdoor data follow distinct triggering paths. It prunes inactive malicious neurons based on clean activation paths to eliminate the backdoor. The pruning criterion for FP is based on the activation values of the Batch Normalization layer in the final network layer, although other activation values could also be used as pruning criteria.

ABL: ABL observes that poison data is more readily learned and often converges earlier in training compared to clean data. It identifies poison samples by discerning differences in loss function values during initial training and then maximizes the loss of these isolated data points to achieve backdoor forgetting.

CLP: CLP operates under the assumption that the backdoor neuron demonstrates higher Channel Lipschitzness. It conducts fine-pruning of the Batch Normalization layer using Channel Lipschitzness to achieve a clean model.

NC: NC uses an optimization searching algorithm to identify a short path within different classes for reverse engineering the backdoor trigger. It then uses the discovered possible trigger to determine whether it's a backdoor model. Finally, it employs fine-tuning to train the model accordingly for mitigation.