ManiBCI: Manipulating EEG BCI with Invisible and Robust Backdoor Attack via Frequency Transform

Anonymous Author(s)
Affiliation
Address
email

Abstract

The electroencephalogram (EEG) based brain-computer interface (BCI) has taken the advantages of the tremendous success of deep learning (DL) models, gaining a wide range of applications. However, DL models have been shown to be vulnerable to backdoor attacks. Although there are extensive successful attacks for image, designing a stealthy and effective attack for EEG is a non-trivial task. While existing EEG attacks mainly focus on single target class attack, and they either require engaging the training stage of the target DL models, or fail to maintain high stealthiness. Addressing these limitations, we exploit a novel backdoor attack called **ManiBCI**, where the adversary can arbitrarily manipulate which target class the EEG BCI will misclassify without engaging the training stage. Specifically, ManiBCI is a three-stages clean label poisoning attacks: **1)** selecting one trigger for each class; **2)** learning optimal injecting EEG electrodes and frequencies masks with reinforcement learning for each trigger; **3)** injecting the corresponding trigger's frequencies into poisoned data for each class by linearly interpolating the spectral amplitude of both data according to the learned masks. Experiments on three EEG datasets demonstrate the effectiveness and robustness of ManiBCI. The proposed ManiBCI also easily bypass existing backdoor defenses. Code will be published after the anonymous period.

1 Introduction

Deep learning (DL) has greatly boosted the performances of the electroencephalogram (EEG) based brain-computer interfaces (BCI), which have been widely used in medical diagnosis [1], healthcare [2], and device control [3, 4]. While DL-based systems are shown to be vulnerable to backdoor attacks (BA) [5–7], where an adversary embeds a hidden backdoor into a DL models to maliciously control it's outputs for inference samples containing particular triggers (a.k.a, poisoned samples), the security of the DL-based EEG BCI has been long neglected.

However, compared to image, designing an effect and stealthy BA for EEG is not trivial for three difficulties, which lead to three questions. **D1**: EEG data has a significantly low signal-to-noise ratio (SNR) [8], even the accuracies of original EEG tasks are very low [9]. **Q1**: How to develop an EEG BA with high attack success rate (ASR) while preserving the clean accuracies of orignial task? **D2**: Previous studies demonstrated for different EEG tasks, there are some different critical EEG electrodes and frequencies that strongly related to the performance of EEG BCI [10–14], indicating that the trigger-injection strategy (*i.e.*, which electrodes and frequencies to inject triggers) inevitably affect the performance of BA. **Q2**: How to find the optimal strategy for different EEG tasks? **D3**: Certain classes of EEG have specific morphology that can easily be identified by human expert, *e.g.*, in epilepsy detection, the amplitudes of the ictal phase EEG are larger than those of the normal state phase EEG [15]. **Q3**: How to maintain the consistency of the label and the morphology?

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

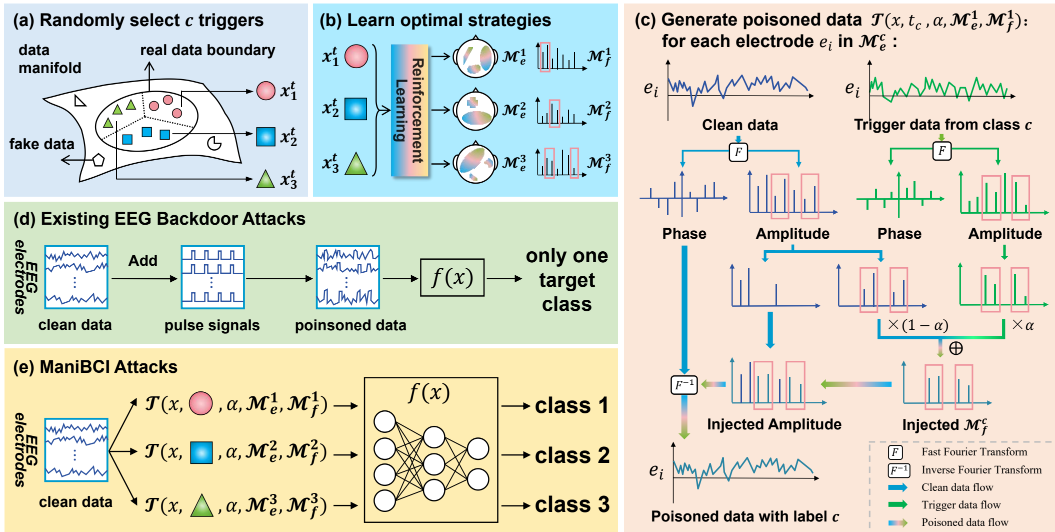


Figure 1: (a)-(c) The framework of ManiBCI: (a) The trigger selection and EEG data distribution from the view of manifold learning. (b) Learning optimal electrodes and frequencies injection strategies. (c) The generation process of ManiBCI. (d) The payloads of the existing backdoor attacks. (e) The payloads of ManiBCI, which can arbitrarily manipulate the output of EEG BCI models.

The first BA for EEG modality is demonstrated in Fig 1 (d), where the narrow period pulse (NPP) signals are added as the trigger for single target class attack [16, 17]. To generate invisible trigger, the adversarial loss is applied to learn a spatial filter as the trigger function [18]. Recently, some BA for time series (EEG signal is a kind of time series) adopt generative adversarial net (GAN) to produce poisoned data [19, 20]. However, there are rich information in the frequency domain of EEG [21–24]. No matter these BA are stealthy or not, they all inject unnatural perturbation in the temporal domain, which will inevitably bring unnatural frequency into the real EEG frequency domain.

In this paper, we propose a novel backdoor attack for **mani**pulating EEG **BCI** called **ManiBCI** to address **Q1**, which injects triggers in the frequency domain. Specifically, ManiBCI is a three-stage clean label poisoning attack demonstrated in Fig 1 (a-c): **1)**: selecting c triggers from c classes , as these triggers are all real EEG, the frequency of these triggers are all natural. Thus, the poisoned data are similar to the real EEG as shown in Fig 2(b). **2)**: learning optimal injecting strategies for each trigger with reinforcement learning to enhance the performance of EEG BA, addressing **Q2**. **3)**: injecting each trigger's frequencies into clean EEG of the same class as the triggers for each class, which maintains the consistency of the label and morphology, addressing **Q3**.

The main contributions of this paper are summarized below:

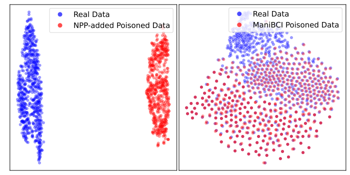
- We propose a novel backdoor attack for EEG BCI called **ManiBCI**, which can attack arbitrary class while preserving stealthiness without engaging the training stage.
- To the best of our knowledge, it is the first work that considers the efficacy of different EEG electrodes and frequencies in EEG backdoor attacks with reinforcement learning.
- Extensive experiments on three EEG BCI datasets demonstrate the effectiveness of ManiBCI and the robustness against several common EEG preprocessings and backdoor defenses.

2 Related Work

2.1 Backdoor Attacks

Backdoor attacks has been deeply investigated in image processing filed [25–27]. BadNets [28] is the first BA, where the adversary maliciously control the DL to misclassify the input images contain suspicious patches to a target class. Other non-stealthy attacks like blended [5] and sinusoidal strips based [29] were studied then. To achieve higher stealthiness, some data poisoning BA were developed, including shifting color spaces [30], warping [31], regularization [32] and frequency-based [33–38]. Other stealthy attacks [39–41] generate invisible trigger patterns by adversarial loss, which requires the control of the model's training process.

Recently, the EEG-based BCIs have shown to be vulnerable to BA [16–18]. The NPP signals are added to clean EEG to generate non-stealthy poisoned samples in [16, 17], which significantly modifies the spectral distribution (as shown in Fig 2 (a)) and results in low stealthiness. From the view of data manifold in Fig 1 (a), NPP-added EEG are fake data. To generate more stealthy poisoned data which stay in the real data boundary. The adversarial loss has been applied backdoor EEG BCI [18] and time series [19, 20], but these methods require controlling the training process of the backdoor models and can only attack a single target class. Meng *et.al.* tried to achieve multi-target attacks with adding different types of signals to clean EEG, *i.e.*, NPP, sawtooth, sine, and chirp [16]. However, these signals are not stealthy in both the temporal and frequency domain. To attack multi-target class with high stealthiness, Marksman backdoor [41] generates invisible sample-specific patterns for each possible class, but it needs controlling the training stage. Moreover, generating trigger patterns with a neural network for each sample is time-consuming.



(a) NPP-based Backdoor Attack (b) ManiBCI Backdoor Attack

Figure 2: t-SNE visualization.

Different from the EEG BA in the temporal domain, we firstly propose to attack in the frequency domain. Our attack is more stealthy than NPP-based attack, faster than other trigger generation attack, and more practical as requiring no control of the target models. It is worth noting that the frequency-based BA for image [33–38] cannot be applied for time series, as they do not consider the characteristics of time series and fail to maintain the stealthiness for poisoned time series data.

2.2 Backdoor Defenses

To cope with the security problems of backdoor attacks, several categories of defensive methods have been developed. Neural Cleanse [42] is a trigger reconstruction based methods. If the reconstructed trigger pattern is significantly small, the model is identified as a backdoor model. Assuming the trigger is still effective when a triggered sample is combining with a clean sample, STRIP [43] detects the backdoor model by feeding the combined samples into the model to see if the predictions are still with low entropy. Spectral Signature [44] detects the backdoor model based on the latent representations. Fine-Pruning [45] erases the backdoor by pruning the model.

Besides the above defenses designed for backdoor attacks, there are some common EEG pre-processing methods, such as bandstop filtering and down-sampling, should be considered when designing a practical robust backdoor attack for EEG BCI in the real-world scene.

3 Methodology

3.1 EEG BCI Backdoor Attacks and Threat Model

Under the supervised learning setting, a classifier f is learned using a labeled training set $\mathcal{S} = \{(x_1, y_1), ..., (x_N, y_N)\}$ to map $f : \mathcal{X} \to \mathcal{C}$, where $x_i \in \mathcal{X}$ and $y_i \in \mathcal{C}$. The attacker in single target class backdoor attacks aims to learn a classifier f behaves as follows:

$$f(x_i) = y_i, \;\; f(T(x_i)) = c_{tar}, \;\; c_{tar} \in \mathcal{C}, \; \forall (x_i, y_i) \in \mathcal{S}, \tag{1}$$

where $T : \mathcal{X} \to \mathcal{X}$ is the trigger function and c_{tar} is the target label. For multi-target class backdoor attacks, the trigger function has an extra parameter c_i, which manipulates the behavior of f flexibly:

$$f(x_i) = y_i, \;\; f(T(c_i, x_i)) = c_i, \;\; \forall c_i \in \mathcal{C}, \forall (x_i, y_i) \in \mathcal{S}. \tag{2}$$

We consider a malicious data provider, who generates a small number of poisoned samples (labeled with the target class) and injects them into the original dataset. A victim developer collects this poisoned dataset and trains his model, which will be infected a backdoor.

We use a cross-validation setting to evaluate all BAs, each EEG dataset $\mathcal{D}$ is divided into three parts: training set $\mathcal{D}_{train}$, poisoning set $\mathcal{D}_p$, and test set $\mathcal{D}_{test}$. Specifically, for a dataset contains n subjects, we select one subject's data as $\mathcal{D}_p$ one by one, and the remaining $n - 1$ subjects to perform leave-one-subject-out (LOSO) cross-validation, *i.e.*, one of the subjects as $\mathcal{D}_{test}$, and the remaining $n-1$ subjects as $\mathcal{D}_{train}$ (one of the subjects in $\mathcal{D}_{train}$ is chosen to be validation set). In summary, for a dataset contains n subjects, there are $n(n-1)$ runs to validate each EEG BCI backdoor attack method. A poisoned subset $\mathcal{S}_p$ of M ($M < N$) examples is generated based on $\mathcal{D}_p$. Then $\mathcal{S}_p$ is combined with $\mathcal{D}_{train}$ to acquire $\mathcal{S} = \{\mathcal{S}_p, \mathcal{D}_{train}\}$. The poisoning ratio is defined as : $\rho = M/N$.

3.2 Reinforcement Learning for Optimal Trigger-Injection Strategies

The learning of the injecting electrodes set $\mathcal{M}_e^{c_i}$ and frequencies set $\mathcal{M}_f^{c_i}$ for each selected trigger in class c_i can be formulated as a non-convex optimization problem. Under this optimization framework, the strategy generator function will learn the optimal $\mathcal{M}_e^{c_i}$ and $\mathcal{M}_f^{c_i}$ for each EEG trigger to implement ManiBCI BA on target DL model f, which is supposed to have a high clean accuracy (CA) on the clean data and attack success rate (ASR) on the poisoned data:

$$\min_{\mathcal{M}_e^{c_i}, \mathcal{M}_f^{c_i}} \mathbb{E}_{(x_i,y_i)\sim\mathcal{D}}[\mathcal{L}(f(x_i), y_i) + \lambda\mathcal{L}(f(\mathcal{T}(x_i, x_{c_i}^t, \alpha, \mathcal{M}_e^{c_i}, \mathcal{M}_f^{c_i})), c_i)]. \tag{3}$$

However, finding the optimal adaptive injecting strategies for each trigger is not trivial as the searching space is too large (*e.g.*, if injecting half of the 62 electrodes, there are $\binom{62}{31} \approx 4.65 \times 10^{17}$ cases for deciding $\mathcal{M}_e^{c_i}$). Reinforcement learning (RL) is an appropriate method for tackling this questions. The objective of RL is to find a sampler π to maximize the expect of the reward function:

$$\begin{aligned} \pi^* &= \arg\max\nolimits_\pi \mathbb{E}_{\tau\sim\pi(\tau)}[R(\tau)] = \arg\max\nolimits_\pi \sum\nolimits_\tau [R(\tau)\cdot p_\pi(\tau)] \\ &= \arg\max\nolimits_\pi \sum\nolimits_\tau [R(\tau)\cdot\rho_0(s_1)\cdot\prod\nolimits_{t=1}^{T-1}\pi(a_t|s_t)\cdot\mathcal{P}(s_{t+1}|s_t,a_t)], \end{aligned} \tag{4}$$

where $R(\tau)$ is reward function of a trajectory $\tau = (s_1, a_1, r_1, ...s_T)$, the s_i, a_i, r_i means the state, action, and reward at time i. The ρ_0 indicates the sampler of initial state. In our settings, the action (strategies) do not affect the state (triggers). Hence, we can simplify Eq 4 by removing the states s_i:

$$\pi^* = \arg\max\nolimits_\pi \sum\nolimits_\tau [R(\tau)\cdot\prod\nolimits_{t=1}^{T-1}\pi(a_t)]. \tag{5}$$

However, we do not care about the reward of the whole trajectory, we only acquire a single strategy for each trigger. Thus, we replace the $R(\tau)$ with $R(a_t)$ and select the a_t whose $R(a_t)$ is the biggest as the optimal strategy. Here, an RL algorithm called policy gradient [46] is adopted to learn an agent (*i.e.*, policy network $\pi_\theta^{c_i}$ with parameters θ) to find the optimal strategy for each trigger. After removing the state s_t and replacing $R(\tau)$, the gradient estimator is:

$$\hat{g} = \nabla_\theta \mathbb{E}_{\tau\sim\pi_\theta(\tau)}[R(\tau)] = \sum\nolimits_\tau [R(a_t)\cdot\nabla p_{\pi_\theta}(a_t)] = \mathbb{E}_t[R_t(a_t)\cdot\nabla_\theta \log\pi_\theta], \tag{6}$$

where a_t and R_t is the action and estimator of the reward function at timestep t. The expectation $\mathbb{E}_t$ indicates the empirical average. Here, $a_t = \{\mathcal{M}_e^{c_i}, \mathcal{M}_f^{c_i}\}$. The parameters of $\pi_\theta^{c_i}$ are updated by $\theta_{t+1} = \theta_t + \eta\hat{g}$, η is the learning rate. We run the RL for T steps and take the best a_t as the strategy.

The CA and ASR are obtained by implementing ManiBCI only on $\mathcal{S}$. Specifically, we use a concise network as the agent which takes the extracted spatial-temporal features from triggers into account to generate better policy. This agent has two output vectors $v_1 \in \mathbb{R}^E, v_2 \in \mathbb{R}^F$, where E and F is the number of EEG electrodes and frequencies. The electrodes and frequencies are in $\mathcal{M}_e^{c_i}$ and $\mathcal{M}_f^{c_i}$ only if the corresponding positions in v_1 and v_2 have Top-k values, k is γE for electrodes and βF for frequencies, where $\gamma, \beta \in (0, 1]$ are hyperparameters.

Besides the performance of CA and ASR, there are two important concerns: **C1:** Robustness against common EEG preprocessig-based defenses; **C2:** Stealthiness against human perceptions. The reason why we consider **C1** is that the bandstop filtering is widely used for preprocessing EEG signals. For instance, if we inject the triggers into a concentrated frequency band 50-60Hz, it is easy to filter the trigger out using a 50Hz low pass filter, resulting in attack failure. Thus, scattering the injection positions in various frequency can effectively evade from specific frequency filter defenses. To address **C2**, injecting the trigger into higher frequencies is more invisible than lower frequencies [47]. Taking all into consideration, we define the estimator of the reward function R_t as follows:

$$R_t(a_t) = R_t(\mathcal{M}_e^{c_i}, \mathcal{M}_f^{c_i}) = \mathrm{CA} + \lambda\,\mathrm{ASR} + \mu\,\mathrm{dis}(\mathcal{M}_f^{c_i}) + \nu\min(\mathcal{M}_f^{c_i}), \tag{7}$$

where the $\mathcal{M}_f^{c_i}$ indicates the set of all injecting frequency positions, and $\mathrm{dis}()$ calculates the minimal distance between each pair of positions. Thus, $\mathrm{dis}(\mathcal{M}_f^{c_i})$ is the discrete (DIS) loss, and $\min(\mathcal{M}_f^{c_i})$ is the high frequency (HF) loss, which can scatter the injection positions in various frequency bands and inject as high frequencies as possible. The $\lambda, \mu, \nu \in \mathbb{R}$ are hyperparameters.

3.3 Poisoned Data Generation via Frequency Transform

After selecting the *C* triggers from each class and learning the strategy for each trigger, the poisoned data are generated by injecting these triggers into clean data with the corresponding strategies. As shown in Fig 1(c), given a clean data $x_i \in \mathcal{D}_p$ with label c_i, and a trigger data $x_{c_i}^t$, let $\mathcal{F}^A$ and $\mathcal{F}^P$ be the amplitude and phase components of the fast Fourier transform (FFT) result of a EEG signals, we denote the amplitude and phase spectrum of x_i and $x_{c_i}^t$ as:

$$\mathcal{A}_{x_i} = \mathcal{F}^A(x_i), \mathcal{A}_{x_{c_i}^t} = \mathcal{F}^A(x_{c_i}^t), \quad \mathcal{P}_{x_i} = \mathcal{F}^P(x_i), \mathcal{P}_{x_{c_i}^t} = \mathcal{F}^P(x_{c_i}^t). \tag{8}$$

The new poisoned amplitude spectrum $\mathcal{A}_{x_i}^P$ is produced by linearly interpolating $\mathcal{A}_{x_i}$ and $\mathcal{A}_{x_{c_i}^t}$. In order to achieve this, we produce a binary mask $\mathcal{M}^{c_i} \in \mathbb{R}^{E \times F} = 1_{(j,k)}, j \in \mathcal{M}_e^{c_i}, k \in \mathcal{M}_f^{c_i}$, whose value is 1 for positions of all corresponding to elements in both electrode and frequency sets and 0 elsewhere. Denoting $\alpha \in (0, 1]$ as the linear interpolating ratio, the new poisoned amplitude spectrum can be computed as follows, where $\odot$ indicates Hadamard product:

$$\mathcal{A}_{x_i}^P = [(1-\alpha)\mathcal{A}_{x_i} + \alpha\mathcal{A}_{x_{c_i}^t}] \odot \mathcal{M}^{c_i} + \mathcal{A}_{x_i} \odot (1 - \mathcal{M}^{c_i}). \tag{9}$$

Finally, we adopt the injected poisoned amplitude spectrum $\mathcal{A}_{x_i}^P$ and the clean phase spectrum $\mathcal{P}_{x_i}$ to get the poisoned data by inverse FFT $\mathcal{F}^{-1}$:

$$x_i^p = \mathcal{F}^{-1}(\mathcal{A}_{x_i}^P, \mathcal{P}_{x_i}). \tag{10}$$

By generating x_i^p through this frequency injection approach, we obtain a subset $\mathcal{S}_p = \{x_1^p, ..., x_M^p\}$, which will combine with $\mathcal{D}_{train}$ to form the whole traing dataset $\mathcal{S}$. The EEG DL model f is then trained with $\mathcal{S}$ to obtain the ability of behvaing as equation 2.

4 Experiments

4.1 Datasets, Baselines, and Experimental Setup

Emotion Recognition (ER) Dataset SEED [12] is a discrete EEG emotion dataset studying three types of emotions: happy, neutral, and sad. SEED collected EEG from 15 subjects.

Motor Imagery (MI) Dataset BCIC-IV-2a [48] dataset recorded EEG from 9 subjects while they were instructed to imagine four types of movements: left hand, right hand, feet, and tongue.

Epilepsy Detection (ED) Dataset CHB-MIT [49] is an epilepsy dataset required from 23 patients. We cropped and resampled the CHB-MIT dataset to build an ED dataset with four types of EEG: ictal, preictal, postictal, and interictal phase EEG.

Non-stealthy Baselines As mentioned in previous sections, to the best of our knowledge, ManiBCI is the first work that studies multi-trigger and multi-target class (MT) backdoor in EEG BCI. For comparison, we design several baseline approaches which can be divided into two main groups: non-stealthy and stealthy. Non-stealthy attacks contains **PatchMT** and **PulseMT**. For a benign EEG segment $x \in \mathbb{R}^{E \times T}$. PatchMT is a multi-trigger and MT extension of BadNets [28] where we fill the first βT timepoints of a EEG segments with a constant number, *e.g.*, {0.1, 0.3, 0.5} for three-class task. PulseMT is a multi-trigger and MT extension of NPP-based backdoor attacks [16] where we use NPP signals with different amplitudes, *e.g.*, {-0.8, -0.3, 0.3, 0.8} for different target classes.

Stealthy Baselines Previous works generate stealthy poisioned samples by controlling the training stage and can only attack single target class [18–20]. As they control the training of target model, it is unfair to directly compare their methods with ManiBCI. There is no stealthy MT BA for EEG. Thus, we design two MT stealthy attacks baselines: **CompMT** and **AdverMT**. CompMT generates poisoned samples for different target classes by compressing the amplitude of EEG with different

Table 1: The clean accuraciy and attack success rate for each target class with 40% poisoning rate. The best results are in **bold** and the second best are underlined.

	Dataset	Emotion Recognition					Motor Imagery						Epilepsy Detection					
	Method	Clean	ASR	0	1	2	Clean	ASR	0	1	2	3	Clean	ASR	0	1	2	3
EEGNet	No Attack	0.477	0.333	-	-	-	0.327	0.250	-	-	-	-	0.508	0.250	-	-	-	-
	PatchMT	0.492	0.382	0.577	0.232	0.337	0.283	0.824	0.866	0.880	0.787	0.762	0.460	0.549	0.532	0.430	0.388	0.845
	PulseMT	0.463	0.778	**0.844**	0.509	**0.981**	0.270	0.825	0.947	0.656	0.758	0.938	0.439	0.810	0.853	0.745	0.729	0.913
	CompMT	0.443	0.385	0.099	0.377	0.678	0.269	0.865	0.530	0.997	0.983	0.948	0.437	0.547	0.261	0.280	0.714	0.933
	AdverMT	0.457	0.334	0.276	0.330	0.396	0.257	0.243	0.316	0.192	0.230	0.235	0.413	0.250	0.326	0.264	0.200	0.210
	ManiBCI	**0.535**	**0.857**	0.831	**0.791**	0.949	**0.323**	**1.000**	**0.999**	**1.000**	**1.000**	**0.999**	**0.477**	**0.944**	**0.930**	**0.954**	**0.921**	**0.970**
DeepCNN	No Attack	0.497	0.333	-	-	-	0.301	0.250	-	-	-	-	0.443	0.250	-	-	-	-
	PatchMT	0.481	0.342	0.248	0.323	0.453	0.276	0.704	0.638	0.977	0.774	0.425	0.431	0.729	0.416	**0.890**	0.719	0.892
	PulseMT	0.450	0.596	**0.815**	0.334	0.638	0.261	0.829	0.764	0.968	0.819	0.765	0.405	**0.885**	**0.872**	0.862	**0.861**	**0.943**
	CompMT	0.461	0.427	0.473	0.473	0.336	0.286	0.887	0.638	0.982	0.946	0.980	0.446	0.538	0.196	0.466	0.571	0.918
	AdverMT	0.367	0.388	0.298	0.453	0.412	0.245	0.247	0.320	0.221	0.196	0.240	0.396	0.275	0.354	0.218	0.227	0.301
	ManiBCI	**0.534**	**0.832**	0.732	**0.865**	**0.901**	**0.315**	**1.000**	**1.000**	**1.000**	**1.000**	**0.999**	**0.469**	0.828	0.725	0.839	0.845	0.904
LSTM	No Attack	0.506	0.333	-	-	-	0.264	0.250	-	-	-	-	0.462	0.250	-	-	-	-
	PatchMT	0.509	0.368	0.311	0.392	0.401	0.261	0.429	0.395	0.296	0.386	0.639	0.450	0.513	0.500	0.437	0.417	0.700
	PulseMT	0.511	0.824	0.883	0.645	0.943	**0.265**	0.533	0.787	0.327	0.282	0.737	0.451	0.804	**0.845**	0.769	0.709	0.895
	CompMT	0.484	0.490	0.272	0.269	0.929	0.260	0.548	0.219	0.511	0.523	0.940	**0.455**	0.435	0.194	0.217	0.490	0.840
	AdverMT	0.367	0.415	0.472	0.453	0.321	0.239	0.271	0.308	0.215	0.247	0.312	0.432	0.268	0.367	0.232	0.198	0.275
	ManiBCI	**0.519**	**0.954**	**0.998**	**0.868**	**0.996**	0.264	**0.966**	**0.987**	**0.988**	**0.901**	**0.986**	0.444	**0.865**	0.795	**0.833**	**0.857**	**0.975**

ratios, *e.g.*, {-0.1, 0, 0.1} for three-class task. AdverseMT is a multi-trigger and MT extension of adversarial filtering based attacks [18], where we using a local model trained only on $\mathcal{S}_p$ to generate different spatial filters $\mathbf{W}_i^*$ for different target classes, then we apply these spatial filters to generate poisoned samples. More details are written in Appendix D.

Experimental Setup We demonstrate the effectiveness of the proposed ManiBCI backdoor through comprehensive experiments on the above three EEG datasets, more details of each dataset and preprocessings are illustrated in Appendix C. We follow the poisoning attack setting as the previous works [16] and consider three EEG DL models for classifier f: EEGNet [50], DeepCNN [51], and LSTM [52, 53]. For all methods, we train the classifiers using the Adam optimizer with learning rate of 0.001. The batch size is 32 and the number of epochs is 100. For all datasets and baselines, the interpolating ratio $\alpha = 0.8$, the electrode poisoning ratio $\beta = 0.1$, the electrode poisoning ratio $\gamma = 0.5$. For the reinforcement learning of ManiBCI, we train π_ξ using the Adam optimizer with learning rate of 0.01. The hyperparameters in advantage function is set to $\lambda = 2$, $\mu = 0.3$, and $\nu = 0.005$. More details of the experimental setup can be found in the supplementary material.

4.2 Effectiveness of ManiBCI

This section presents the attack success rates of ManiBCI and baselines. To evaluate the performance in the multi-trigger multi-payload scenario, for each test sample $(x, y) \in \mathcal{D}_{test}$, we enumerate all possible target labels $c_i \in \mathcal{C}$ including the true label y and inject the trigger to activate the backdoor. The attack is successful only when the backdoor classifier f correctly predicts c_i for each poisoned input x with a target label c_i.

4.2.1 Attack Performance

The clean-data accuracy (Clean) and ASR (Attack) for each class of all attack methods on three EEG tasks with three EEG DL models are presented in Table 1. The AdverMT, designed for single-target attack, fails to attacks multiple target classes. Our ManiBCI significantly outperforms baselines at almost all cases ($p < 0.05$) except attacking DeepCNN on the ED dataset, having ASRs above 0.8 on three datasets and even achieving an ASR of 1.000 on the MI dataset. These results demonstrate that our ManiBCI is effective across different EEG tasks and EEG models. PulseMT achieves the second best on ER and ED dataset, CompMT achieves the second best on the MI dataset.

4.2.2 Performance of the Reinforcement Learning: Policy Gradient

Displaying in Table 2, the performance of the policy gradient was compared with other common optimazation algorithms, including genetic algorithm (GA) [54] and random selection (The search space is too large for performing grid search as explained in Section 3.2). It can be observed that the policy gradient outperforms GA while only spending 16% training time of GA. We plot the learning curve of RL in Appendix F.3, which demonstrates that RL learns well strategies within 50 epochs, i.e., only trains 50 backdoor models and saves lots of time. The random algorithm can achieve a not bad results, proving that our methods can be applied without RL if some performance drop is acceptable.

Table 2: Clean and attack performance with with different trigger search optimization algorithms, the poisoning rate is set to 10%. The target model is EEGNet.

Method \ Dataset	Emotion			Motor Imagery			Epilepsy		
	Clean	Attack	Time ↓	Clean	Attack	Time ↓	Clean	Attack	Time ↓
Random	0.520	0.771	-	0.291	0.857	-	0.501	0.721	-
Genetic Algorithm	0.516	0.826	15.2h	0.302	1.000	10.0h	0.492	0.862	30.5h
Policy Gradient	0.535	0.857	2.5h	0.323	1.000	1.8h	0.477	0.944	5.2h

4.2.3 Performance of Learned Mask Strategies on Other Target Models

We demonstrate that the injecting strategies learned on a EEG classifier f can be used to attack other EEG classifiers $\hat{f}$. In other words, Marksman can still be effective when the adversary has no knowledge of the target models $\hat{f}$. To perform the experiments, we use the strategy learned with a classifier f, then generate poisoned samples to attack another classifier $\hat{f}$ whose network is different from f. Table 3 shows the performance difference, it can be observed that the difference is relatively small in most of the cases, demonstrating the transferability of the injecting strategy learned with reinforcement learning.

Table 3: Clean and attack performance on other models. Red values represent the decreasing performance in attacks with f is the same as $\hat{f}$. Blue values mean increments or unchanged .

Models	f : EEGNet				f : DeepCNN				f : LSTM			
	$\hat{f}$: DeepCNN		$\hat{f}$: LSTM		$\hat{f}$: EEGNet		$\hat{f}$: LSTM		$\hat{f}$: EEGNet		$\hat{f}$: DeepCNN	
Datasets	Clean	Attack	Clean	Attack	Clean	Attack	Clean	Attack	Clean	Attack	Clean	Attack
Emotion	0.458 0.026	0.781 0.051	0.485 0.034	0.938 0.016	0.516 0.019	0.813 0.044	0.490 0.029	0.936 0.018	0.516 0.019	0.863 0.006	0.497 0.037	0.779 0.053
Motor	0.316 0.001	1.000 0.000	0.265 0.001	0.946 0.020	0.309 0.014	1.000 0.000	0.264 0.000	0.972 0.006	0.306 0.017	1.000 0.000	0.306 0.009	1.000 0.000
Epilepsy	0.442 0.027	0.759 0.069	0.469 0.025	0.806 0.059	0.448 0.029	0.943 0.001	0.445 0.001	0.813 0.052	0.448 0.029	0.926 0.018	0.427 0.042	0.850 0.022

4.2.4 Attack Performance with Different Hyperparameters

We investigate the influences of three different hyperparameters: poisoning rate ρ, frequency injection rate β, and electrode injection rate γ. The performance of attacking EEGNet on the ED dataset are displayed in Fig 3. It can be seen that the ASRs are positively correlated with poisoning rate. Note that it is non-trivial for multi-target class attack, thus the ASR is not high compared to the single class attack. ManiBCI outperforms other attacks in all cases and is robust to the change of β and γ.

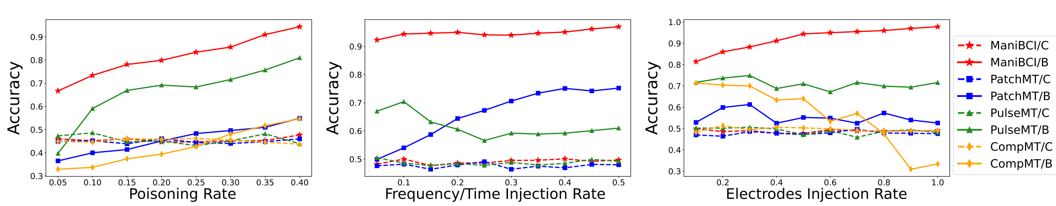


Figure 3: Clean (/C) and attack (/B) performance with different poisoning or injection rates.

4.3 Robustness of ManiBCI

In this section, we evaluate the robustness of our ManiBCI against different EEG preprocessing method and various representative backdoor defenses.

4.3.1 Robustness against EEG Preprocessing Methods

To develop an EEG BCI, it is very common to preprocess the raw EEG signals, *e.g.*, 1) band-stop filtering and 2) down-sampling. An EEG backdoor attack is impractical in real scenarios if it is no longer effective when the target model is trained with the preprocessed poisoned EEG. Hence, we must take the robustness against preprocessing methods into account, which is widely ignored in the image backdoor attack field. The performance of each method facing different preprocessing methods are presented in Table 4. It can be observed that our ManiBCI is robust in all cases. However, when removing the DIS loss, the performance of ManiBCI decreases a lot after EEG preprocessing, especially facing the 30 Hz high-stop filtering preprocessing due to the HF loss that encourages the policy network learns to injecting high frequency.

Table 4: Clean and attack performance on three datasets after different EEG preprocessing methods. The target model is EEGNet. M w.o. DIS means removing the DIS loss in ManiBCI.

	Preprocessing	No defense		20 Hz low		30 Hz high		25% down		Average
	Method	Clean	Attack	Clean	Attack	Clean	Attack	Clean	Attack	ASR
ER	ManiBCI	0.535	0.857	0.512	0.829	0.463	0.892	0.518	0.908	0.876
	w/o DIS	0.506	0.859	0.492	0.816	0.466	0.333	0.498	0.807	0.652
MI	ManiBCI	0.323	1.000	0.285	1.000	0.329	1.000	0.321	1.000	1.000
	w/o DIS	0.298	1.000	0.264	1.000	0.322	0.250	0.284	0.990	0.746
ED	ManiBCI	0.497	0.944	0.492	0.914	0.494	0.856	0.516	0.818	0.920
	w/o DIS	0.515	0.250	0.477	0.864	0.508	0.250	0.510	0.249	0.454

4.3.2 Robustness against Neural Cleanse: Trigger Inversion

Neural Cleanse (NC) [42] calculate a metric called Anomaly Index by reconstructing trigger pattern for each possible label. The Anomaly Index is positively correlated with the size of the reconstruction trigger. A model with Anomaly Index > 2 is considered to be backdoor-injected. We display the Anomaly Indexes of the clean models and the backdoor-injected model by ManiBCI in Fig 4. It can be seen that ManiBCI can easily bypass NC. The reconstructed trigger patterns on three datasets are presented in Appendix F.1.

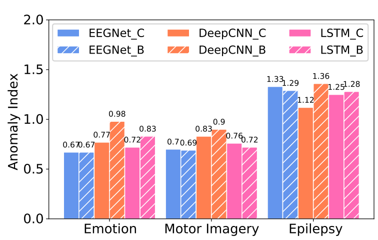


Figure 4: Anomaly Index of three models on three datasets.

4.3.3 Robustness against STRIP: Input Perturbation

We evaluate the robustness of ManiBCI against STRIP [43], which perturbs the input EEG and calculates the entropy of the predictions of these perturbed EEG data. Based on the assumption that the trigger is still effective after perturbation, the entropy of backdoor input tends to be lower than that of the clean one. The results are plotted in Fig 5, it can be seen that the entropy distributions of the backdoor and clean samples are similar.

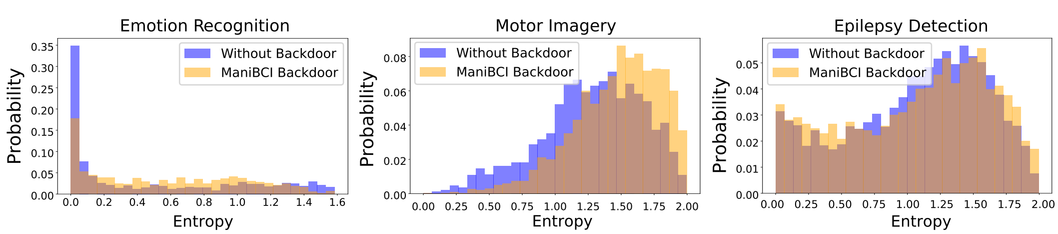


Figure 5: Performance against STRIP on three datasets, the target model is EEGNet.

4.3.4 Robustness against Spectral Signature: Latent Space Correlation

Spectral Signature [44] detects the backdoor samples by statistical analysis of clean data and backdoor data in the latent space. Following the same experimental settings in [44], we randomly select 5,000 clean samples and 500 ManiBCI backdoor samples and plot the histograms of the correlation scores in Fig 6. There is no clear separation between these two sets of samples, showing the stealthiness of ManiBCI backdoor samples in the latent space.

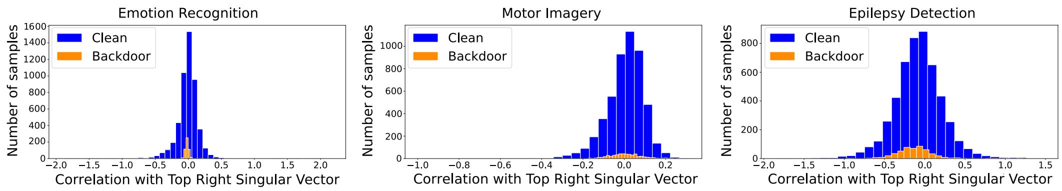


Figure 6: Performance against Spectral Signature on three datasets, the target model is EEGNet.

4.3.5 Robustness against Fine-Pruning

We evaluate the robustness of Marksman against Fine-Pruning [45], a model analysis based defense which finds a classifier's low-activated neurons given a small clean dataset. Then it gradually prunes these low-activated neurons to mitigate the backdoor without affecting the CA. We can observe from Fig 7 that the ASR drops considerably small when pruning ratio is less than 0.7, suggesting that the Fine-Pruning is ineffective against ManiBCI.

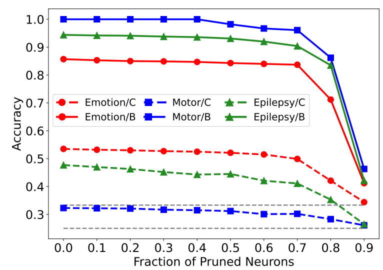


Figure 7: Performances of EEG-Net against Fine-Pruning on three datasets.

4.4 Visualization of Backdoor Attack Samples

To evade from human perception (**C2** in Section 3.2), we design to obatin injecting strategies with HF loss. It can be seen from the bottom row of Fig 8 that ManiBCI (with HF loss) generates stealthy poisoned EEG, which is almost the same as the clean EEG, demonstrating the **High Stealthiness**. The poisoned EEG will be conspicuous compared to the clean EEG if remove the HF loss.

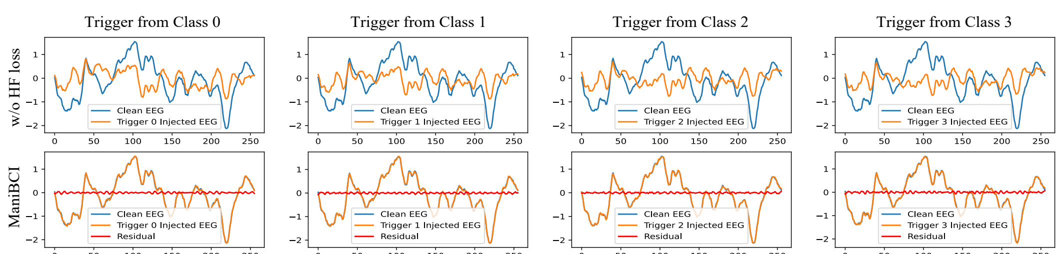


Figure 8: The Clean EEG (Blue), Trigger-injected EEG (Orange) and the Residual (Red) of the ED dataset. The x-axis is the timepoints, the y-axis is the normalized amplitude. Top row: w.o. HF loss; Bottom row: with HF loss. Each column indicates each possible class.

5 Conclusion

In this paper, we proposed ManiBCI, a novel EEG backdoor for manipulating EEG BCI, where the adversary can arbitrarily control the output for any input samples. To the best of our knowledge, ManiBCI is the first method that considers which EEG electrodes and frequencies to be injected by adopting a reinforcement learning called policy gradient to learn the adaptive injecting strategies for different EEG triggers and tasks. We specially design the reward function in RL to enhance the robustness and stealthiness of ManiBCI. The perturbation of the trigger on clean EEG is almost invisible. Our experimental results over three common EEG datasets demonstrate the effectiveness of ManibCI and the stealthiness against the existing representative defenses. This work calls for defensive studies to counter ManiBCI for EEG modality.

References

[1] I. Ahmad, X. Wang, M. Zhu, C. Wang, Y. Pi, J. A. Khan, S. Khan, O. W. Samuel, S. Chen, G. Li *et al.*, "EEG-based epileptic seizure detection via machine/deep learning approaches: a systematic review," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[2] M. Jafari, A. Shoeibi, M. Khodatars, S. Bagherzadeh, A. Shalbaf, D. L. García, J. M. Gorriz, and U. R. Acharya, "Emotion recognition in EEG signals using deep learning methods: A review," *Computers in Biology and Medicine*, p. 107450, 2023.

[3] H. Lorach, A. Galvez, V. Spagnolo, F. Martel, S. Karakas, N. Intering, M. Vat, O. Faivre, C. Harte, S. Komi *et al.*, "Walking naturally after spinal cord injury using a brain–spine interface," *Nature*, vol. 618, no. 7963, pp. 126–133, 2023.

[4] H. Altaheri, G. Muhammad, M. Alsulaiman, S. U. Amin, G. A. Altuwaijri, W. Abdul, M. A. Bencherif, and M. Faisal, "Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: A review," *Neural Computing and Applications*, vol. 35, no. 20, pp. 14 681–14 722, 2023.

[5] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, "Targeted backdoor attacks on deep learning systems using data poisoning," *arXiv preprint arXiv:1712.05526*, 2017.

[6] Y. Gao, B. G. Doan, Z. Zhang, S. Ma, J. Zhang, A. Fu, S. Nepal, and H. Kim, "Backdoor attacks and countermeasures on deep learning: A comprehensive review," *arXiv preprint arXiv:2007.10760*, 2020.

[7] R. Shokri *et al.*, "Bypassing backdoor detection algorithms in deep learning," in *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2020, pp. 175–183.

[8] S. L. Kappel, D. Looney, D. P. Mandic, and P. Kidmose, "Physiological artifacts in scalp EEG and ear-EEG," *Biomedical Engineering Online*, vol. 16, pp. 1–16, 2017.

[9] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. J. Miller, G. R. Müller-Putz *et al.*, "Review of the bci competition iv," *Frontiers in Neuroscience*, vol. 6, p. 55, 2012.

[10] M. Z. Parvez and M. Paul, "EEG signal classification using frequency band analysis towards epileptic seizure prediction," in *16th Int'l Conf. Computer and Information Technology*. IEEE, 2014, pp. 126–130.

[11] R. Jana and I. Mukherjee, "Deep learning based efficient epileptic seizure prediction with EEG channel optimization," *Biomedical Signal Processing and Control*, vol. 68, p. 102767, 2021.

[12] W.-L. Zheng and B.-L. Lu, "Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks," *IEEE Transactions on Autonomous Mental Development*, vol. 7, no. 3, pp. 162–175, 2015.

[13] M. Z. Baig, N. Aslam, and H. P. Shum, "Filtering techniques for channel selection in motor imagery EEG applications: a survey," *Artificial Intelligence Review*, vol. 53, no. 2, pp. 1207–1232, 2020.

[14] P. Herman, G. Prasad, T. M. McGinnity, and D. Coyle, "Comparative analysis of spectral approaches to feature extraction for EEG-based motor imagery classification," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 16, no. 4, pp. 317–326, 2008.

[15] W. T. Blume, G. B. Young, and J. F. Lemieux, "EEG morphology of partial epileptic seizures," *Electroencephalography and Clinical Neurophysiology*, vol. 57, no. 4, pp. 295–302, 1984.

[16] L. Meng, X. Jiang, J. Huang, Z. Zeng, S. Yu, T.-P. Jung, C.-T. Lin, R. Chavarriaga, and D. Wu, "EEG-based brain-computer interfaces are vulnerable to backdoor attacks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2023.

[17] X. Jiang, L. Meng, S. Li, and D. Wu, "Active poisoning: efficient backdoor attacks on transfer learning-based brain-computer interfaces," *Science China Information Sciences*, vol. 66, no. 8, p. 182402, 2023.

[18] L. Meng, X. Jiang, X. Chen, W. Liu, H. Luo, and D. Wu, "Adversarial filtering based evasion and backdoor attacks to EEG-based brain-computer interfaces," *Information Fusion*, p. 102316, 2024.

[19] D. Ding, M. Zhang, Y. Huang, X. Pan, F. Feng, E. Jiang, and M. Yang, "Towards backdoor attack on deep learning based time series classification," in *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2022, pp. 1274–1287.

[20] Y. Jiang, X. Ma, S. M. Erfani, and J. Bailey, "Backdoor attacks on time series: A generative approach," in *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2023, pp. 392–403.

[21] S. Arroyo and S. Uematsu, "High-frequency eeg activity at the start of seizures," *Journal of Clinical Neurophysiology*, vol. 9, no. 3, pp. 441–448, 1992.

[22] M. Kostyunina and M. Kulikov, "Frequency characteristics of eeg spectra in the emotions," *Neuroscience and Behavioral Physiology*, vol. 26, no. 4, pp. 340–343, 1996.

[23] M. Salinsky, B. Oken, and L. Morehead, "Test-retest reliability in eeg frequency analysis," *Electroencephalography and clinical neurophysiology*, vol. 79, no. 5, pp. 382–392, 1991.

[24] S. D. Muthukumaraswamy, "High-frequency brain activity and muscle artifacts in meg/eeg: a review and recommendations," *Frontiers in human neuroscience*, vol. 7, p. 138, 2013.

[25] M. Weber, X. Xu, B. Karlaš, C. Zhang, and B. Li, "Rab: Provable robustness against backdoor attacks," in *2023 IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2023, pp. 1311–1328.

[26] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 12 250–12 259.

[27] Z. Yuan, P. Zhou, K. Zou, and Y. Cheng, "You are catching my attention: Are vision transformers bad learners under backdoor attacks?" in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 24 605–24 615.

[28] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg, "BadNets: Evaluating backdooring attacks on deep neural networks," *IEEE Access*, vol. 7, pp. 47 230–47 244, 2019.

[29] M. Barni, K. Kallas, and B. Tondi, "A new backdoor attack in cnns by training set corruption without label poisoning," in *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 101–105.

[30] W. Jiang, H. Li, G. Xu, and T. Zhang, "Color backdoor: A robust poisoning attack in color space," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 8133–8142.

[31] T. A. Nguyen and A. T. Tran, "Wanet-imperceptible warping-based backdoor attack," in *International Conference on Learning Representations (ICLR)*, 2020.

[32] 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 Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2088–2105, 2020.

[33] Y. Zeng, W. Park, Z. M. Mao, and R. Jia, "Rethinking the backdoor attacks' triggers: A frequency perspective," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 16 473–16 481.

[34] T. Wang, Y. Yao, F. Xu, S. An, H. Tong, and T. Wang, "An invisible black-box backdoor attack through frequency domain," in *European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 396–413.

[35] H. A. A. K. Hammoud and B. Ghanem, "Check your other door! creating backdoor attacks in the frequency domain," *arXiv preprint arXiv:2109.05507*, 2021.

[36] R. Hou, T. Huang, H. Yan, L. Ke, and W. Tang, "A stealthy and robust backdoor attack via frequency domain transform," *World Wide Web (WWW)*, vol. 26, no. 5, pp. 2767–2783, 2023.

[37] Y. Feng, B. Ma, J. Zhang, S. Zhao, Y. Xia, and D. Tao, "Fiba: Frequency-injection based backdoor attack in medical image analysis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 20 876–20 885.

[38] 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 *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 38, no. 3, 2024, pp. 1851–1859.

[39] T. A. Nguyen and A. Tran, "Input-aware dynamic backdoor attack," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 3454–3464, 2020.

[40] K. Doan, Y. Lao, W. Zhao, and P. Li, "Lira: Learnable, imperceptible and robust backdoor attacks," in *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, 2021, pp. 11 966–11 976.

[41] K. D. Doan, Y. Lao, and P. Li, "Marksman backdoor: Backdoor attacks with arbitrary target class," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, pp. 38 260–38 273, 2022.

[42] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, "Neural Cleanse: Identifying and mitigating backdoor attacks in neural networks," in *2019 IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2019, pp. 707–723.

[43] 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 *Proceedings of the 35th Annual Computer Security Applications Conference*, 2019, pp. 113–125.

[44] B. Tran, J. Li, and A. Madry, "Spectral signatures in backdoor attacks," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.

[45] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *International Symposium on Research in Attacks, Intrusions, and Defenses*. Springer, 2018, pp. 273–294.

[46] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 12, 1999.

[47] S. V. Gliske, Z. T. Irwin, K. A. Davis, K. Sahaya, C. Chestek, and W. C. Stacey, "Universal automated high frequency oscillation detector for real-time, long term eeg," *Clinical Neurophysiology*, vol. 127, no. 2, pp. 1057–1066, 2016.

[48] C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, and G. Pfurtscheller, "BCI competition 2008–graz data set A," *Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology*, vol. 16, pp. 1–6, 2008.

[49] A. H. Shoeb and J. V. Guttag, "Application of machine learning to epileptic seizure detection," in *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010, pp. 975–982.

[50] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, 2018.

[51] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for EEG decoding and visualization," *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.

[52] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[53] K. M. Tsiouris, V. C. Pezoulas, M. Zervakis, S. Konitsiotis, D. D. Koutsouris, and D. I. Fotiadis, "A long short-term memory deep learning network for the prediction of epileptic seizures using eeg signals," *Computers in biology and medicine*, vol. 99, pp. 24–37, 2018.

[54] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimedia tools and applications*, vol. 80, pp. 8091–8126, 2021.

Appendix

A Limitations

Our ManiBCI is a backdoor attack in the frequency domain, which requires to transform the EEG signals into frequency domain through fast Fourier transform (FFT) and return to temporal domain through inverse FFT (iFFT). The operation of FFT and iFFT in the trigger injection function are a little more time-consuming compared to other backdoor attack directly in the temporal domain, like PatchMT [28] and PulseMT [16]. Future effort will be devoted into the faster implementation of FFT and iFFT, for example, taking the advantage of modern GPUs.

It is a little more time-consuming for the reinforcement learning to acquire the optimal strategies for each trigger. However, we can obtain a general injecting strategy for each EEG BCI tasks, which can achieve a relatively good performance without reinforcement learning, as we can see from Table 3 that random injection strategy has an acceptable performance.

B Broader Impacts

With the rapid development of techniques, EEG BCIs gain a wide range of applications from health care to human-computer interaction. Some companies like Neuralink adopt the EEG BCI to assist paralytic patients helping themselves in daily lives. However, if the EEG BCI is backdoor attacked by ManiBCI, which allows the attacker to arbitrarily control BCI's outputs, the BCI users may fall into tremendous fatal troubles. For instance, one paralytic patient controls his/her wheelchair by EEG BCI, the attacker can manipulate the wheelchair to run down a steep staircase. For an epileptic patient, the attacker can let all the output be Normal State, even when the patient is experiencing an epileptic seizure. This paper reveals the severe danger faced by EEG BCIs, demonstrating the possibility that someone can maliciously manipulate the outputs of EEG BCIs with arbitrary target class.

ManiBCI can also be used for positive purposes, like protecting intellectual property of EEG dataset and EEG models with watermarking. As our ManiBCI has a very small impact of the clean accuracy, and the poisoning approach is clean label poisoning, ManiBCI is a fantastic method for watermarking EEG dataset and models.

For a company that provides EEG dataset, it can select different EEG triggers for different customs to generate poisoned data and inject into the dataset provided to customs who buy the dataset. As a result, the company have the information of which trigger is corresponding to which customs, e.g., trigger x is in the dataset provided to custom X, trigger y is in the dataset provided to custom Y. If an EEG model from a company which didn't buy dataset is detected having this watermark (backdoor) with trigger x, the company knows that the custom X leaked the dataset. Similarly, if an EEG model is detected having this watermark (backdoor) with trigger y, the company knows that the custom Y leaked the dataset.

C Datasets and Preprocessing

In this section, we introduce the three datasets used in our experiments, and explain the preprocessing. Table 5 presents some basic information of these datasets.

Table 5: Basic information of the three datasets

Dataset	Emotion	Motor Imagery	Epilepsy
Class Numbers	3	4	4
Subjects	15	9	23
Electrodes	62	22	23
Sampling Rate	200 Hz	250 Hz	256 Hz

C.1 Emotion Recognition (ER)

The SJTU Emotion EEG Dataset (SEED) was incoporated as the representative dataset of emotion recogniton tasks [12]. It consists of EEG recordings from 15 subjects watching 15 emotional video clips with three repeated session each on different days. Each video clip is supposed to evoke one of the three target emotions: positive, neutral, and negative. The EEG signals were acquired by the 62-channel electrode cap at a sampling rate of 1000 Hz. We performed below preprocessing procedures for the 62-channel EEG signals: 1) Down-sampling from 1000 Hz to 200 Hz, 2) Band-pass filtering at 0.3-50 Hz, 3) Segmenting EEG signals into 1-second (200 timepoints), obtaining 3394 EEG segments in each session for each subject.

C.2 Motor Imagery (MI)

We employ the BCIC-IV-2a as a representative dataset of MI classification tasks [48]. It contains EEG recordings in a four-class motor-imagery task from nine subjects with two repeated session each on different days. During the task, the subjects were instructed to imagine four types of movements (*i.e.*, right hand, left hand, feet, and tongue) for four seconds. Each session consists of a total of 288 trials with 72 trials for each type of the motor imagery. The EEG signals were recorded by 22 Ag/AgCl EEG electrodes in a sampling rate of 250 Hz. We segment the 22-channel EEG signals into 1-second segments, resulting in totally 1152 EEG data for each subject.

C.3 Epilepsy Detection (ED)

The CHB-MIT, one of the largest and most used public datasets for epilepsy, is adopted as a representative dataset of ED tasks [49]. It recorded 877.39 hours of multi-channel EEG in a sampling rate of 256 Hz from 23 pediatric patients with intractable seizures. However, as the montages (*i.e.*, the number and the places of electrodes) of EEG signals vary significantly among different subjects' recordings, we select to use only the EEG recordings with the same 23 channels (see Appendix A) and discard other channels or the recordings don't have all these 23 channels. Due to the purpose is to test whether the backdoor attack works on the ED task, not to study the epilepsy EEG classification, we segment part of the CHB-MIT dataset to form a four-class ED dataset (*i.e.*, the preictal, ictal, postictal, and interictal phases). Specifically, for a ictal phase EEG recording of t_i seconds from $[s_i, e_i]$ timepoints, we segment the $[s_i - t_i, e_i]$ EEG as the preictal phase, the $[e_i, e_i + t_i]$ EEG as the postictal phase, and another t_i seconds EEG recordings as the interictal phase which satisfying there is no ictal phase within half an hour before or after. Then we segment the 23-channel EEG signals into 1-second segments, consequently, there are 41336 segments left in total from all subjects, 10334 for each phase. As the imbalanced amount of data across different subjects, we separate these 41336 segments into 10 groups and treat the ten groups as 10 subjects.

D Implementation Details

D.1 Experiment Computing Resources

We use two servers for conducting our experiments. A server with one Nvidia Tesla V100 GPU is used for running reinforcement learning, the CUDA version is 12.3. Another server with four Nvidia RTX 3090 GPUs is used for running the backdoor attacks, the CUDA version is 11.4.

D.2 Details of Baseline Methods

In our ManiBCI backdoor attacks, for an EEG segment $x_i \in \mathbb{R}^{E \times T}$, we modify the βF frequency-points and γE electrodes of a EEG segments with a constant number.

There are four baseline methods in our study for multi-target backdoor attacks, two of them are non-stealthy attacks (**PatchMT** and **PulseMT**) and two are stealthy attacks (**CompressMT** and **AdverseMT**). In order to achieve a fair comparison, we modify only first γE electrodes for all baseline attack methods. For the non-stealthy attacks, which are all on the temporal domains, we modify βT timepoints of EEG signals. For the stealthy attacks, there is no constraint of the numbers of the modify timepoints as these attacks achieve stealthiness in another way.

For each baseline method, we try our best to find out the best performance, as demonstrated below. We promise that we did not maliciously lower the performances of the baseline methods.

D.2.1 PatchMT

PatchMT is a multi-trigger and MT extension of BadNets [28] where we fill the first βT timepoints and γE electrodes of a EEG segments with a constant number. Specifically, for an EEG segment $x_i \in \mathbb{R}^{E\times T}$, we set the first γE electrodes and the first βT timepoints of the EEG segment to a constant number. We normalize the EEG segment $x_i \in \mathbb{R}^{E\times T}$ to let $\mathbf{x}_i$'s mean is 0 and std is 1. Then set the first γE electrodes and the first βT timepoints of $\mathbf{x}_i$ to a different constant number for different class. The constant number for each class of $\{0, 1, 2, 3\}$ for four classes, and {-0.1, 0.0, 1.0} for three classes. Finally, denormalize $\mathbf{x}_i$ to original signal x_i's scale to generate x_i^p.

Although we try our best to find the best performance of PatchMT, and BadNets [28] is really efficient in image backdoor attacks, PatchMT cannot have satisfactory results in EEG BCI attack.

D.2.2 PulseMT

For PulseMT, we met the same questions as the PatchMT: how to identify the amplitude of each NPP signal for each class? If the numbers are too large then normal EEG signals, it will be unfair. If the numbers are too small, the efficacy of PulseMT is too negative.

We normalize the EEG segment $x_i \in \mathbb{R}^{E\times T}$ to let $\mathbf{x}_i$'s mean is 0 and std is 1. The constant amplitude for each class of $\{-0.8, -0.3, 0.3, 0.8\}$. Finally, denormalize $\mathbf{x}_i$ to original signal x_i's scale to generate x_i^p.

D.2.3 CompressMT

Compressing the amplitude of EEG signals in the temporal domain will not change the morphology and the frequency distribution of EEG signals, thus obtaining stealthiness. For three-class Emotion datasets, the compress rate is {0.8, 0.6, 0.4}. For four-class Motor Imagery and Epilepsy datasets, the compress rate is {0.8, 0.6, 0.4, 0.2}.

D.2.4 AdverseMT

AdverseMT is another stealthy EEG backdoor attacks, which is the multi-trigger and multi-target extension of adversarial spatial filter attacks [18], in wihch, for EEG segment $x_i \in \mathbb{R}^{E\times T}$, it learns an Spatial Filter $\mathbf{W} \in \mathbb{R}^{E\times E}$ by the adversarial loss to let the model f misclassify x_i:

$$\min_{\mathbf{W}} \mathbb{E}_{(x_i,y_i)\sim\mathcal{D}}[-\mathcal{L}_{CE}(\mathbf{W}x_i, y_i) + \alpha\mathcal{L}_{MSE}(\mathbf{W}x_i, x_i)], \tag{11}$$

However, the original version of [18] requires the access to all training dataset $\mathcal{D}$ and the control of the training process of the model f. We modify the AdverseMT to only access to the training dataset $\mathcal{D}_{train}$. Note that the adversarial loss dose not have the special design for multi-target backdoor attacks, we only run the process c times for obtaining c spatial filters for different classes. So the poisoned subset are $\mathcal{S}_p = \{(\mathbf{W}_0(x), 0), (\mathbf{W}_1(x), 1), (\mathbf{W}_2(x), 2), (\mathbf{W}_3(x), 3)\}$.

D.3 Reinforcement Learning Policy Network Architecture

Here, we design a concise but effective convolutional neural networks as the our policy network, which is defined as belows:

Table 6: The Architecture of Policy Network

Layer	In	Out	Kernel	Stride
Conv2d	1	32	(1, 3)	(1, 1)
BatchNorm2d				
ELU				
AvgPool2d				(1,2)
Conv2d	32	64	(1, 3)	(1, 1)
BatchNorm2d				
ELU				
AvgPool2d				(1,2)
AdaptiveAvgPool2d				(1, 1)
Flatten				
Linear	64	256		

E Attack Performance of ManiBCI

E.1 Different Poisoning Rates

We present the performance of each backdoor attacks' performance under different poisoning rates in Table 7. We can see that our ManiBCI outperforms other baseline at all poisoning rates, demonstrating the superiority of ManiBCI. Note that the performance of ManiBCI on the MI dataset is significantly robust to low poisoning rates, i.e., ASR of 1.000 when $\rho = 0.05$.

E.2 Hyperparameter Analysis: Frequency and Electrodes Injection Ratio

We present the performance of each backdoor attacks performance under different rates in Table 8 and Table 9. It can be observed with the increment of β and γ, the attack performance increases. Because the trigger is bigger in clean EEG data.

E.3 Hyperparameter Analysis in Reinforcement Learning

We applied the following reward function to acquire the optimal mask strategies for each triggers:

$$Q_t = \mathrm{CA} + \lambda\,\mathrm{ASR} + \mu\,\mathrm{dis}(\mathcal{M}_f^{c_i}) + \nu\min(\mathcal{M}_f^{c_i}), \tag{12}$$

where the first part means the clean accuracy, the second part means the attack success rate, the third part is aiming to scatter the injection positions in various frequency bands, and the fourth part is aiming to inject as high frequencies in EEG signals as possible. Here, we give a simple example to demonstrate the reward function. For an 10 timepoints long EEG segment x_i, $\widetilde{x}_i = \mathcal{F}(x_i)$. If the $\mathcal{M}_f^{c_i} = \{2, 3, 5, 7, 9\}$, because the minimal distance between each pair in $\mathcal{M}_f^{c_i}$ is $|2 - 3| = 1$, thus $\mathrm{dis}(\mathcal{M}_f^{c_i}) = 1$. The $\min(\mathcal{M}_f^{c_i})$ means the lowest position in $\mathcal{M}_f^{c_i}$, thus $\min(\mathcal{M}_f^{c_i}) = 2$.

The analysis of the λ are presented in Table 10. When λ increase, the Attack performance increases while the Clean performance declines slightly.

Table 10: Clean (/C) and attack (/B) performance with ASR's hyperparameter λ, $\mu = 0.3, \nu = 0.005$

	Dataset	Emotion		Motor Imagery		Epilepsy	
	Method	Clean	Attack	Clean	Attack	Clean	Attack
0.5	ManiBCI	0.542±0.03	0.847±0.04	0.327±0.02	1.000±0.01	0.500±0.04	0.922±0.04
1.0	ManiBCI	0.537±0.02	0.855±0.03	0.325±0.02	1.000±0.01	0.482±0.03	0.935±0.05
2	ManiBCI	0.535±0.03	0.857±0.02	0.323±0.02	1.000±0.01	0.477±0.04	0.944±0.02

Table 7: Clean (/C) and attack (/B) performance with different poisoning rates for ManiBCI and other baseline methods. The target model is EEGNet for all cases.

ρ	Dataset	Emotion		Motor Imagery		Epilepsy	
	Method	Clean	Attack	Clean	Attack	Clean	Attack
0.05	PatchMT	0.390	0.333	0.281	0.791	0.449	0.365
	PulseMT	0.488	0.337	0.275	0.788	0.473	0.397
	ComprsMT	0.448	0.313	0.269	0.754	0.449	0.329
	ManiBCI	0.491	0.566	0.321	1.000	0.460	0.667
0.10	PatchMT	0.443	0.334	0.279	0.785	0.452	0.400
	PulseMT	0.445	0.394	0.281	0.796	0.486	0.591
	ComprsMT	0.509	0.323	0.270	0.778	0.446	0.337
	ManiBCI	0.541	0.718	0.320	1.000	0.452	0.734
0.15	PatchMT	0.455	0.335	0.285	0.805	0.439	0.414
	PulseMT	0.438	0.514	0.280	0.787	0.447	0.669
	ComprsMT	0.488	0.332	0.275	0.792	0.461	0.374
	ManiBCI	0.528	0.805	0.322	1.000	0.460	0.781
0.20	PatchMT	0.481	0.334	0.277	0.816	0.461	0.451
	PulseMT	0.447	0.555	0.285	0.810	0.451	0.692
	ComprsMT	0.470	0.347	0.270	0.795	0.458	0.394
	ManiBCI	0.538	0.773	0.321	1.000	0.447	0.799
0.25	PatchMT	0.487	0.335	0.281	0.820	0.444	0.483
	PulseMT	0.466	0.701	0.275	0.815	0.431	0.684
	ComprsMT	0.493	0.335	0.269	0.800	0.462	0.427
	ManiBCI	0.551	0.836	0.325	1.000	0.447	0.834
0.30	PatchMT	0.459	0.343	0.280	0.809	0.440	0.496
	PulseMT	0.486	0.810	0.272	0.816	0.451	0.716
	ComprsMT	0.499	0.331	0.269	0.825	0.455	0.481
	ManiBCI	0.526	0.829	0.320	1.000	0.451	0.756
0.35	PatchMT	0.437	0.341	0.285	0.805	0.448	0.510
	PulseMT	0.437	0.767	0.275	0.837	0.482	0.757
	ComprsMT	0.473	0.347	0.265	0.851	0.446	0.517
	ManiBCI	0.489	0.763	0.321	1.000	0.453	0.910
0.40	PatchMT	0.490	0.345	0.283	0.824	0.460	0.549
	PulseMT	0.454	0.771	0.270	0.825	0.439	0.443
	ComprsMT	0.464	0.361	0.269	0.865	0.437	0.450
	ManiBCI	0.528	0.849	0.323	1.000	0.477	0.944

Table 8: Clean (/C) and attack (/B) performance with frequency injection rate β, $\gamma = 0.5$

β	Dataset	Emotion		Motor Imagery		Epilepsy	
	Method	Clean	Attack	Clean	Attack	Clean	Attack
0.05	PatchMT	0.411	0.334	0.272	0.801	0.476	0.499
	PulseMT	0.464	0.752	0.265	0.800	0.505	0.670
	ManiBCI	0.522	0.744	0.319	0.999	0.482	0.923
0.10	PatchMT	0.431	0.363	0.283	0.824	0.482	0.540
	PulseMT	0.460	0.795	0.270	0.825	0.486	0.704
	ManiBCI	0.522	0.813	0.323	1.000	0.500	0.944
0.15	PatchMT	0.413	0.371	0.275	0.821	0.464	0.587
	PulseMT	0.449	0.701	0.271	0.821	0.477	0.632
	ManiBCI	0.532	0.848	0.322	0.998	0.477	0.947
0.20	PatchMT	0.390	0.377	0.271	0.829	0.479	0.644
	PulseMT	0.434	0.769	0.270	0.819	0.484	0.606
	ManiBCI	0.529	0.882	0.325	0.999	0.486	0.950
0.25	PatchMT	0.406	0.385	0.267	0.835	0.491	0.673
	PulseMT	0.491	0.705	0.275	0.832	0.478	0.566
	ManiBCI	0.519	0.865	0.328	0.999	0.486	0.941
0.30	PatchMT	0.417	0.382	0.269	0.831	0.464	0.706
	PulseMT	0.425	0.708	0.273	0.844	0.488	0.592
	ManiBCI	0.521	0.862	0.330	0.999	0.495	0.940
0.35	PatchMT	0.435	0.373	0.270	0.841	0.475	0.734
	PulseMT	0.423	0.621	0.276	0.839	0.479	0.589
	ManiBCI	0.527	0.850	0.332	0.998	0.496	0.947
0.40	PatchMT	0.438	0.378	0.271	0.843	0.469	0.751
	PulseMT	0.481	0.624	0.272	0.845	0.485	0.592
	ManiBCI	0.521	0.893	0.330	0.999	0.501	0.951
0.45	PatchMT	0.460	0.385	0.266	0.844	0.481	0.742
	PulseMT	0.429	0.633	0.277	0.856	0.499	0.601
	ManiBCI	0.519	0.877	0.325	0.999	0.492	0.962
0.50	PatchMT	0.423	0.386	0.263	0.840	0.480	0.752
	PulseMT	0.459	0.514	0.273	0.851	0.492	0.610
	ManiBCI	0.528	0.893	0.329	1.000	0.497	0.970

Table 9: Clean (/C) and attack (/B) performance with electrodes injection rate γ, $\beta = 0.1$

γ	Dataset	Emotion		Motor Imagery		Epilepsy	
	Method	Clean	Attack	Clean	Attack	Clean	Attack
0.10	PatchMT	0.431	0.334	0.268	0.795	0.470	0.529
	PulseMT	0.425	0.498	0.269	0.802	0.502	0.717
	ComprsMT	0.407	0.349	0.271	0.805	0.482	0.656
	ManiBCI	0.489	0.485	0.235	0.367	0.499	0.814
0.20	PatchMT	0.473	0.335	0.271	0.805	0.464	0.599
	PulseMT	0.469	0.707	0.270	0.816	0.502	0.737
	ComprsMT	0.465	0.363	0.268	0.812	0.514	0.704
	ManiBCI	0.481	0.709	0.235	0.367	0.486	0.860
0.30	PatchMT	0.423	0.343	0.272	0.803	0.486	0.613
	PulseMT	0.488	0.767	0.273	0.814	0.506	0.749
	ComprsMT	0.451	0.398	0.271	0.811	0.494	0.700
	ManiBCI	0.500	0.743	0.235	0.367	0.490	0.883
0.40	PatchMT	0.453	0.343	0.270	0.812	0.478	0.525
	PulseMT	0.467	0.786	0.271	0.816	0.498	0.688
	ComprsMT	0.443	0.361	0.270	0.820	0.506	0.634
	ManiBCI	0.491	0.767	0.235	0.367	0.478	0.912
0.50	PatchMT	0.431	0.363	0.270	0.813	0.472	0.552
	PulseMT	0.460	0.795	0.269	0.819	0.471	0.710
	ComprsMT	0.430	0.366	0.269	0.821	0.503	0.640
	ManiBCI	0.522	0.813	0.235	0.367	0.477	0.944
0.60	PatchMT	0.452	0.377	0.267	0.819	0.480	0.549
	PulseMT	0.460	0.808	0.269	0.823	0.490	0.672
	ComprsMT	0.459	0.368	0.271	0.826	0.499	0.534
	ManiBCI	0.488	0.828	0.235	0.367	0.495	0.950
0.70	PatchMT	0.443	0.368	0.272	0.812	0.497	0.525
	PulseMT	0.437	0.809	0.270	0.821	0.459	0.716
	ComprsMT	0.456	0.366	0.273	0.835	0.492	0.571
	ManiBCI	0.527	0.853	0.235	0.367	0.489	0.955
0.80	PatchMT	0.461	0.383	0.268	0.821	0.479	0.573
	PulseMT	0.456	0.771	0.267	0.829	0.488	0.699
	ComprsMT	0.431	0.383	0.270	0.833	0.488	0.475
	ManiBCI	0.539	0.865	0.235	0.367	0.489	0.960
0.90	PatchMT	0.439	0.400	0.271	0.817	0.478	0.540
	PulseMT	0.461	0.811	0.269	0.823	0.494	0.694
	ComprsMT	0.459	0.389	0.274	0.836	0.490	0.309
	ManiBCI	0.520	0.824	0.235	0.367	0.489	0.970
1.00	PatchMT	0.430	0.370	0.267	0.823	0.476	0.526
	PulseMT	0.456	0.794	0.271	0.829	0.482	0.716
	ComprsMT	0.453	0.376	0.269	0.830	0.490	0.334
	ManiBCI	0.532	0.846	0.235	0.367	0.491	0.978

F More Visualization Results

In this section, we plot the reconstructed triggers and masks on three datasets in Section F.1, then plot more visualizations of backdoor samples in Section **??**, and plot the learning curve of our reinforcement learning in Section F.3.

F.1 Neural Cleanse: Reconstruction Trigger Patterns

Here, we present more visualization in Figure 9, Figure 10, and Figure 11 of the reconstructed trigger patterns and mask patterns for each possible label on three dataset (*i.e.*, the CHB-MIT dataset, the BCIC-IV-2a dataset and the SEED dataset) the target model is EEGnet. It can be observed that the reconstructed trigger patterns and mask patterns of the clean models and ManiBCI backdoor-injected models are very similar to each other. Thus, our ManiBCI backdoor attack can easily bypass the defense of Neural Cleanse.

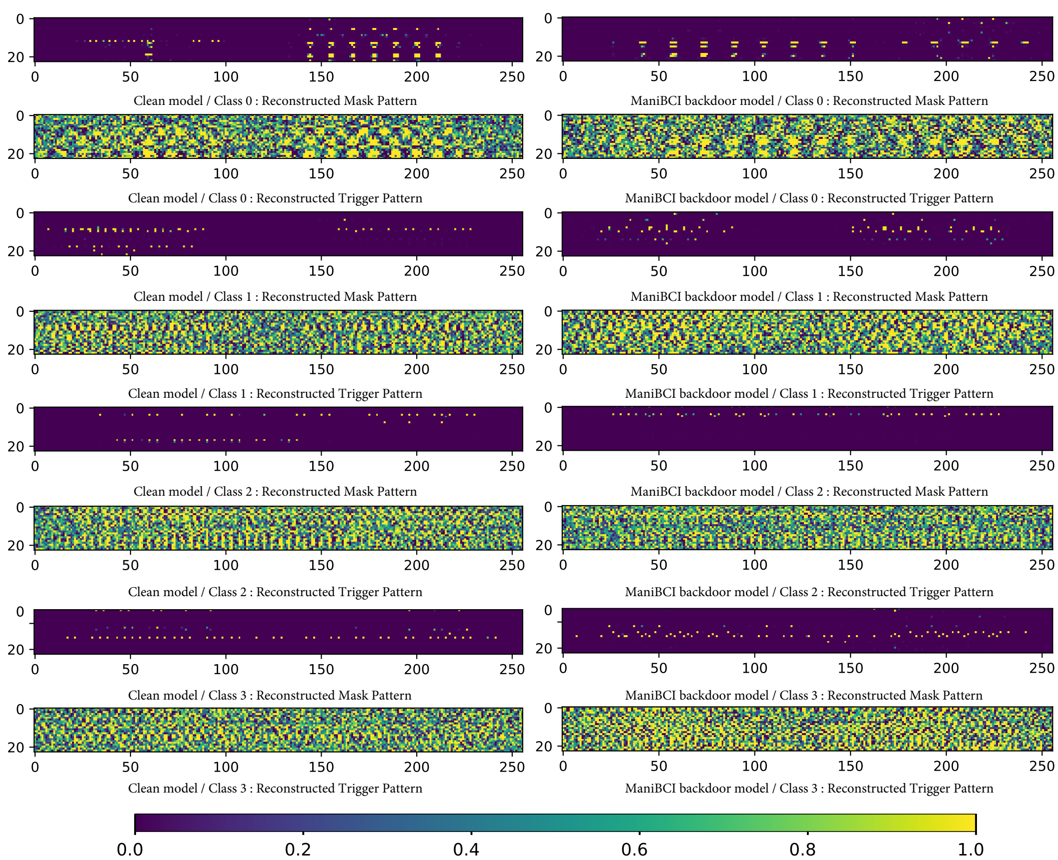


Figure 9: The reconstructed trigger patterns and mask patterns for each possible class in the CHB-MIT dataset. The results in the left column are reconstructed based on the clean model, the results in the right column are reconstructed based on the backdoor model. The EEG segments in the CHB-MIT dataset have 23 electrodes and 256 timepoints.

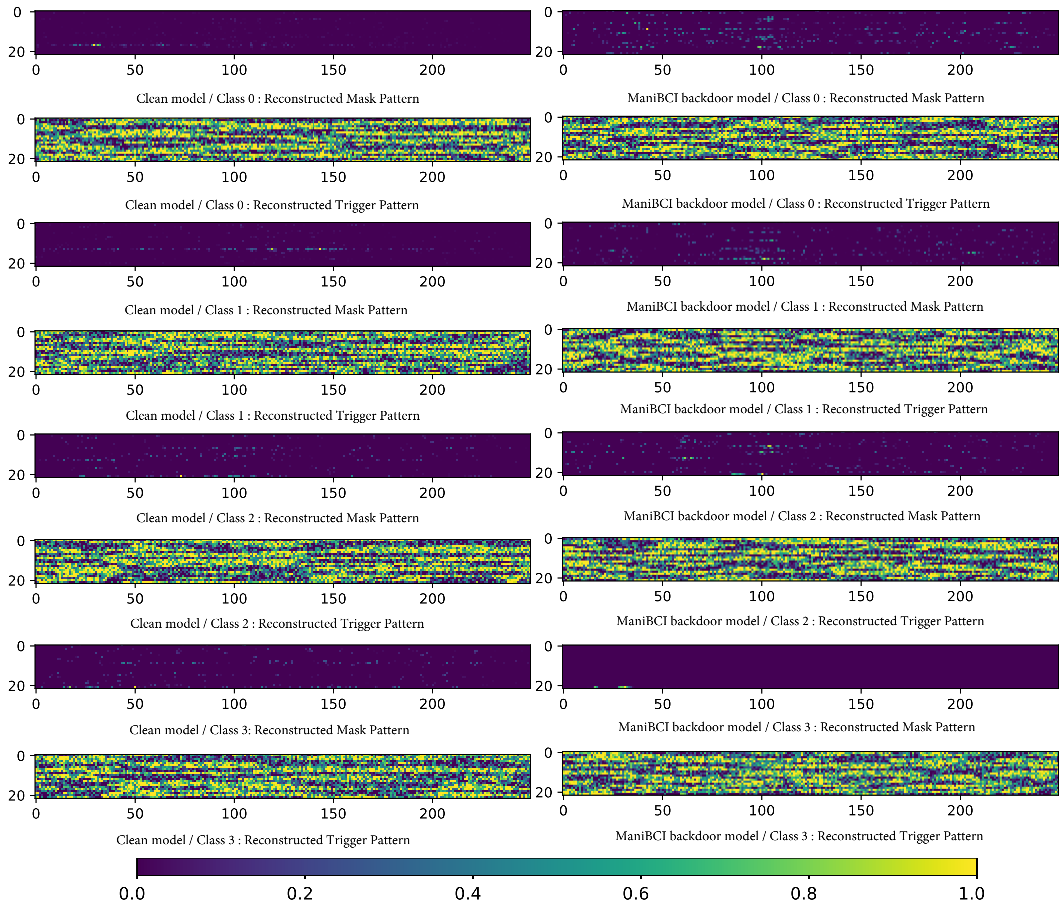


Figure 10: The reconstructed trigger patterns and mask patterns for each possible class in the MI dataset. The results in the left column are reconstructed based on the clean model, the results in the right column are reconstructed based on the backdoor model. The EEG segments in the MI dataset have 22 electrodes and 250 timepoints.

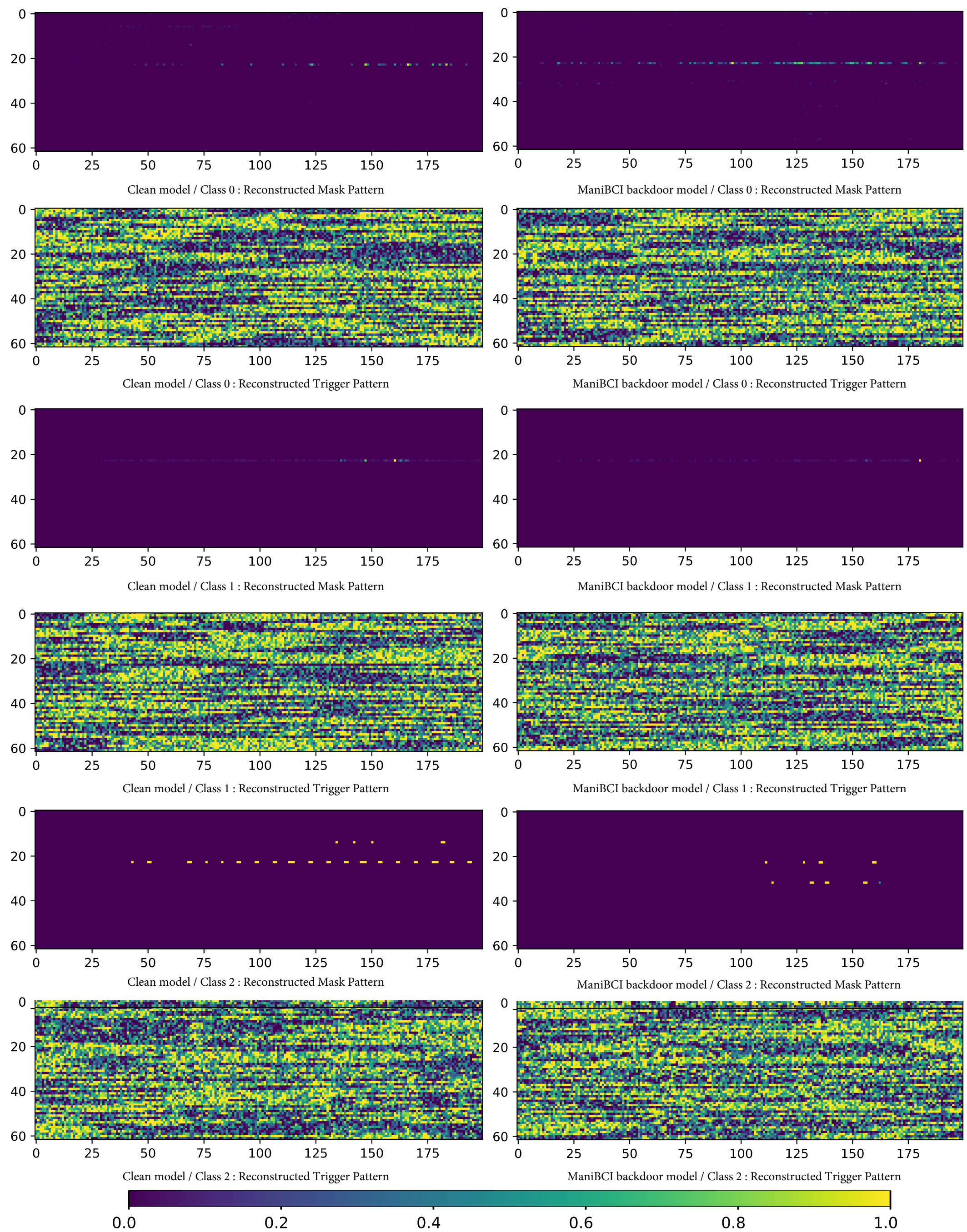


Figure 11: The reconstructed trigger patterns and mask patterns for each possible class in the ER dataset (i.e., SEED dataset). The results in the left column are reconstructed based on the clean model, the results in the right column are reconstructed based on the backdoor model. The EEG segments in the SEED dataset have 62 electrodes and 200 timepoints.

F.2 Visualization of Backdoor Attack Samples

We present more visualization of the backdoor attack samples generated by our ManiBCI on ER dataset and MI dataset in Fig 12 and 13. The x-axis is the timepoints, the y-axis is the normalized amplitude. Top row: w.o. HF loss; Bottom row: with HF loss. Each column indicates each possible class.

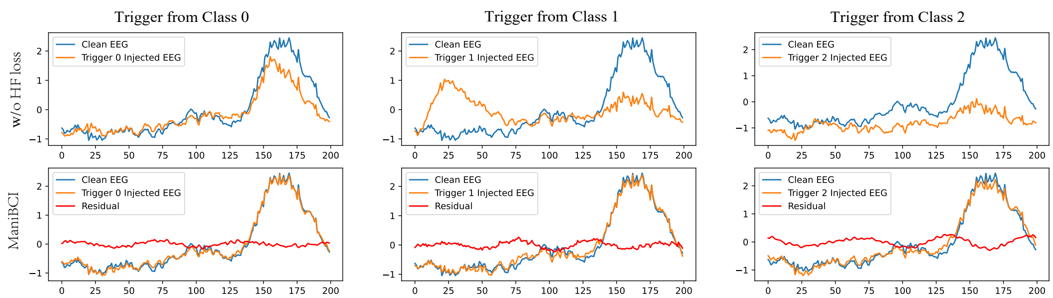


Figure 12: The Clean EEG (Blue), Trigger-injected EEG (Orange) and the Residual (Red) of the ER dataset.

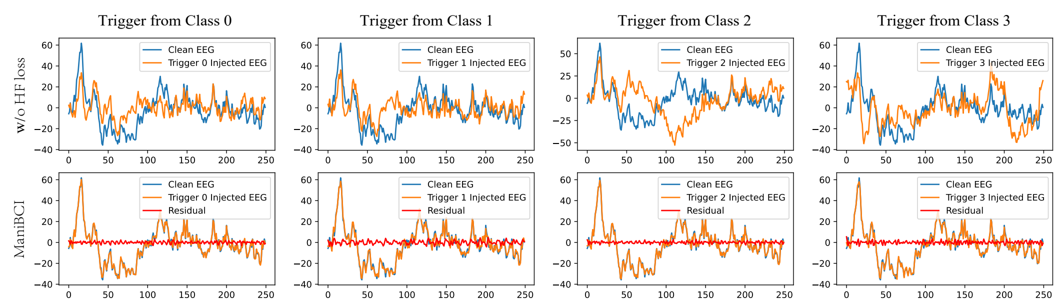
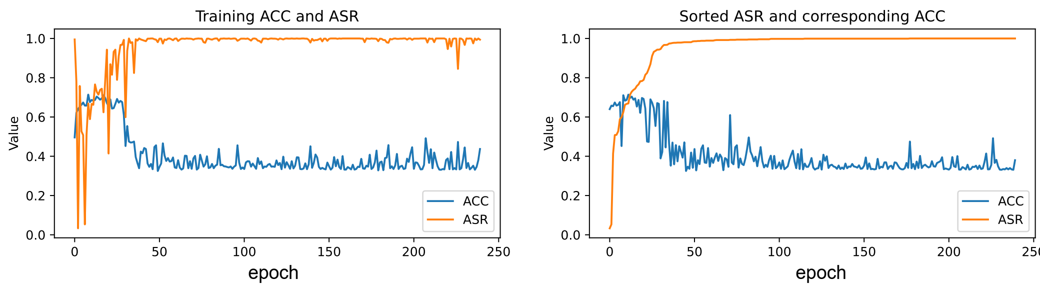


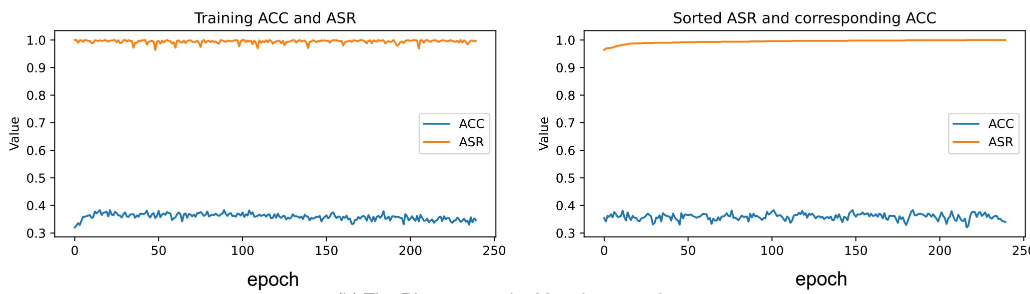
Figure 13: The Clean EEG (Blue), Trigger-injected EEG (Orange) and the Residual (Red) of the MI dataset.

F.3 Visualization of Learning Curves of Reinforcement Learning

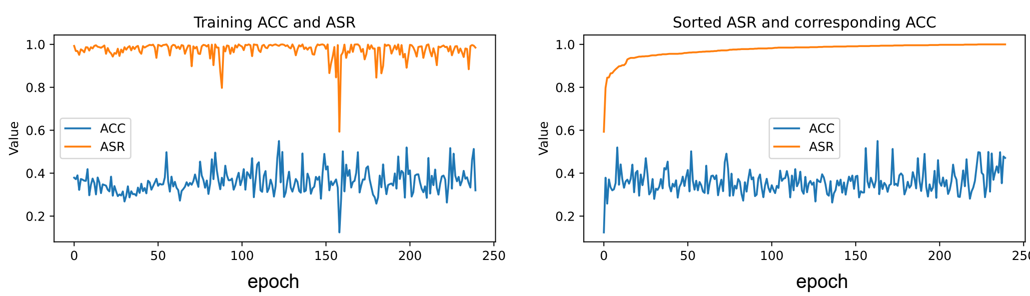
We present the visualization of the learning curves of the reinforcement learning of three dataset in Fig 14. We can see the effectiveness of our reinforcement, which converged within 50 epochs on the ER dataset, that is, only trained 50 backdoor models with different injection strategies. Our RL is more effective on the MI dataset and ED dataset, which finds a good strategy within less 10 epochs. Our RL is robust when learning strategies for different triggers as demonstrated in Fig 14(c) and (d), where the learning curves are quite similar when RL is performing on different triggers.



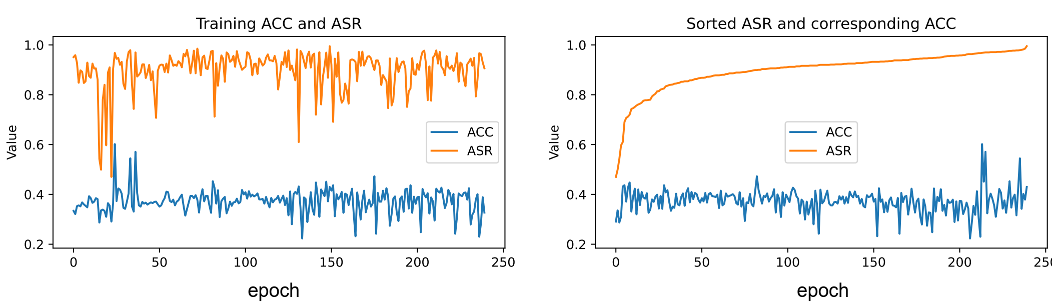
(a) The RL curve on the Emotion Recognition dataset



(b) The RL curve on the Moto Imagery dataset



(c) The RL curve on the Epilepsy Detection dataset



(d) The RL curve on the Epilepsy Detection dataset, for another tirrger with label 2

Figure 14: The learning curves of RL on three datasets. The right column is the curve we sort the (ACC,ASR) according to the ASR. The backdoor models are all EEGNet.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We made clear claims of our contributions in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discussed the limitations of our proposed method in Appendix.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our paper dose not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We demonstrated our method and the experiment settings clearly in Section 3.1 and Section 4.2. The implementation details of all baselines are written in appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Sorry for not providing the whole code at the submitting phase as we have no time to organize our code well. However, we will publish our code after the anonymous period (Or we can organize and upload our code during rebuttal phase if possible).

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We demonstrated our method and the experiment settings clearly in Section3.1 and Section 4.2. The implementation details of all baselines are written in appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We give all the statistical significance of our experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Yes, we provide the type of GPU and version of CUDA in Appendix D.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, we conform with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the broader impacts of our backdoor attacks in Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We do not release any dataset or model. However, our paper proposes a backdoor attack method in EEG BCIs, which is challenging to be guarded and have dangerous impact in EEG BCIs. The only safeguard way we can come up with is to check and guarantee the clean of training datasets EEG BCIs employ.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We conduct our experiments on three public datasets. The original papers of these three datasets were cited in our paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.