BACKDOOR ATTACKS IN THE SUPPLY CHAIN OF MASKED IMAGE MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked image modeling (MIM) revolutionizes self-supervised learning (SSL) for image pre-training. In contrast to previous dominating self-supervised methods, i.e., contrastive learning, MIM attains state-of-the-art performance by masking and reconstructing random patches of the input image. However, the associated security and privacy risks of this novel generative method are unexplored. In this paper, we perform the first security risk quantification of MIM through the lens of backdoor attacks. Different from previous work, we are the first to systematically threat modeling on SSL in every phase of the model supply chain, i.e., pre-training, release, and downstream phases. Our evaluation shows that models built with MIM are vulnerable to existing backdoor attacks in release and downstream phases and are compromised by our proposed method in pre-training phase. For instance, on CIFAR10, the attack success rate can reach 99.62%, 96.48%, and 98.89% in the downstream phase, release phase, and pre-training phase, respectively. We also take the first step to investigate the success factors of backdoor attacks in the pre-training phase and find the trigger number and trigger pattern play key roles in the success of backdoor attacks while trigger location has only tiny effects. In the end, our empirical study of the defense mechanisms across three detection-level on model supply chain phases indicates that different defenses are suitable for backdoor attacks in different phases. However, backdoor attacks in the release phase cannot be detected by all three detection-level methods, calling for more effective defenses in future research.

1 INTRODUCTION

The self-supervised pre-training task has been dominant by contrastive learning, a discriminative method, in the computer vision domain since 2018 (Zhang et al., 2022). Recently, with the advent of the Transformer architecture, masked image modeling (MIM), a generative method, has successfully surpassed contrastive learning and reached state-of-the-art performance on self-supervised pre-training tasks (Bao et al., 2021; He et al., 2021; Chen et al., 2022; Xie et al., 2021). Compared with contrastive learning which aims to align different augmented views of the same image, MIM learns from predicting properties of masked patches from unmasked parts. It plays as a milestone that bridges the gap between visual and linguistic self-supervised pre-training methods, and has quickly emerged variants in applications such as images (An et al., 2022; Bachmann et al., 2022), video (Wei et al., 2021; Tong et al., 2022), audio (Baade et al., 2022), and graph (Tan et al., 2022). However, as an iconic method settling in another branch of SSL, the associated security risks caused by the mask-and-predict mechanism and novel architectures of MIM are still unexplored.

Our Contributions. In this paper, we perform the first security risk quantification of MIM through the lens of backdoor attacks. Different from previous work, we are the first to systematically categorize the threat models on MIM in every phase of model supply chain, i.e., pre-training, release, and downstream phases (see Section 3 for more details). Our evaluation shows that models built with MIM are vulnerable to existing backdoor attacks in release and downstream phases. For instance, in the downstream phase, with only 0.1% poisoning rate (e.g., only 50 training samples on CIFAR10) and 0.05% occupied area of the image, the attacker can achieve 89.37% ASR on CIFAR10.

We also observe that previous attack (Saha et al., 2021), which successfully backdoors contrastive learning in the pre-training phase, cannot achieve satisfying attack performance on MIM. The ASR is only 2.83% and 13.78% higher than the baseline on CIFAR10 and STL10, respectively. To improve the attack performance in the pre-training phase, we propose a simple yet effective method: increasing the number of triggers in the span of the whole image. We observe that, with our method, the ASR rises to 98.89% and 97.74% on CIFAR10 and STL10 datasets, respectively.

To further investigate the hardest yet rarely explored scenario, i.e., the pre-training phase, we conduct comprehensive ablation studies on the properties of triggers, i.e., pattern, location, number, size, and poisoning rate. We find that trigger pattern and trigger number are key components that affect attack performance on MIM, which is different from a previous study on contrastive learning (Saha et al., 2021). We utilize the white trigger and publicly released triggers of Hidden Trigger Backdoor Attacks (HTBA) to evaluate the effects of trigger pattern (Saha et al., 2020). We observe that the white triggers only get 7.19% ASR on STL10, while the ASRs of trigger HTBA-10, HTBA-12, and HTBA-14 are 97.74%, 98.05%, 62.74%.

Our fourth contribution is the empirical study of the defense mechanisms. Concretely, we investigate the detection performance from three detection-level on all model supply chain phases. Our evaluation shows that both model-level (Wang et al., 2019) and input-level (Gao et al., 2019) defenses can detect backdoor attacks in the downstream phase while dataset-level (Tran et al., 2018) defense works well in recognizing poisoned samples in the pre-training dataset. To our surprise, backdoor attacks in the release phase, called Type II attack in our paper, cannot be detected by all three detection-level methods, which prompts the call for more effective defenses in future research.

2 Preliminary

2.1 Masked Image Modeling (MIM)

The core idea of MIM is masking random parts of the image and then learning to reconstruct the missing parts. It follows the autoencoder design with the transformer architecture as the building blocks to perform the task. The input image is first cropped to patches, e.g., 16 × 16 patches, and MIM randomly masks certain portions of patches. The encoder then maps the unmasked patches to a latent representation and uses the decoder to predict properties of masked patches from the latent representation. The predicted property can be the original pixels (He et al., 2021), latent representation (Wei et al., 2021), or visual tokens (Bao et al., 2021; Chen et al., 2022). The objective of MIM is to minimize the difference between predicted properties and real properties of masked patches. Generally speaking, MIM can be concluded into two categories, tokenizer-based methods (Chen et al., 2022) and end-to-end methods (Zhang et al., 2022).

Tokenizer-Based MIM. Inspired by the success of masked language modeling, tokenizer-based MIM mimics BERT (Devlin et al., 2019) to reconstruct visual tokens. It includes two steps: utilizes an image tokenizer to generate tokens of masked patches and then optimizes the loss by predicting the correct tokens via visual patches.

End-to-End MIM. As the name implies, end-to-end MIM is a one-stage method without the pre-trained tokenizer. The method is straightforward and effective. By directly predicting large portions of masked patches with the help of small portions of unmasked patches, it can achieve impressive performance.

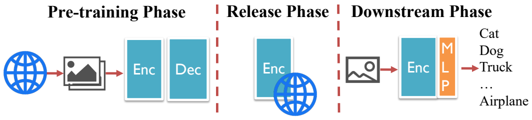


Figure 1: Supply chain of self-supervised models.

2.2 Supply Chain of Self-Supervised Models

As Figure 1 displays, the supply chain of self-supervised models can be generally summarized into three phases. The first phase is the pre-training phase, where the model owner utilizes images collected by the data donor to train the self-supervised model. The second phase is the release phase where the model owner makes the trained model available online via public platforms such as ModelZoo [1] and Hugging Face [2]. The third phase is the downstream phase. In this phase, the

[1] https://modelzoo.co/

[2] https://huggingface.co/

Table 1: Attack Taxonomy. The attacks are increasingly harder in row order. ●: Applicable or Necessary; ○: Inapplicable or Unnecessary; ◐: Partially Applicable or Necessary

Phase →	Pre-training	Release	Downstream		
Attack ↓, Capability →	Pre-training set	Model	Downstream set	Downstream model	Inference pipeline
Type I	○	○	◐	○	○
Type II	○	●	○	○	○
Type III	◐	○	○	○	○

downstream model owner adopts the pre-trained encoder as the backbone and fine-tunes an extra classification layer, i.e., MLP layer, to perform the downstream tasks. The new model (containing an encoder and a classifier) is called *downstream model.*

2.3 Backdoor Attacks

In general, backdoor attacks inject hidden backdoors into machine learning models so that the infected models perform well on clean images but misclassify images with a specific trigger into a target class. As an emerging and rapidly growing research area, various backdoor attacks have been proposed (Gu et al., 2017; Carlini & Terzis, 2021; Saha et al., 2021; 2020; Wang et al., 2020; Liu et al., 2020; Jia et al., 2022) and can be broadly summarized into two categories, i.e., poisoning-based and non-poisoning-based backdoor attacks (Li et al., 2020).

Poisoning-based Backdoor Attack. Give a training set $(X, Y) \in \mathcal{D}_{train}$, we first denote a target model as $f : X \to Y$ where $X \subset \mathbb{R}^d$ is a set of data samples and $Y = \{1, 2, ..., K\}$ is a set of labels. Given a sample x with its label y, we assume the adversary has a target label $\widetilde{y}$ and a trigger patch t. The attacker constructs a poisoned pair $(\widetilde{x}, \widetilde{y})$ by replacing the label y to $\widetilde{y}$ and pasting the trigger t on the image x to get the patched image $\widetilde{x}$. Then, the attacker injects a portion p of poisoned pair $(\widetilde{x}, \widetilde{y})$ into $\mathcal{D}_{train}$ ($0 < p < 1$). Since the victim is not aware that the training set has been modified, the backdoor would be successfully embedded in the model after the training process.

Non-poisoning-based Backdoor Attacks. Different from poisoning-based backdoor attacks, non-poisoning-based backdoor attacks (Rakin et al., 2020; Jia et al., 2022) directly modify model parameters to inject backdoors without poisoning the training set. Normally, given a clean model f, the attacker aims to optimize it to a backdoored model f'. Concretely, the attacker collects a shadow dataset $\mathcal{D}_{shadow}$ poisoned with trigger t and adopt a reference image r from the target class $\widetilde{y}$. The optimization problem aims to minimize the distance between $\mathcal{D}_{shadow}$ and r.

3 Attack Taxonomy and Methodology

As we are the first to investigate backdoor attacks on masked image modeling, we begin by defining our adversary's goal with a unified attack taxonomy covering all phases in the model supply chain. Note, the attack taxonomy can also be generally extended to self-supervised models.

Adversary's Goal. Following previous work (Gu et al., 2017; Jia et al., 2022), we assume the adversary aims to backdoor the downstream model so that the model performs well on clean images but misclassifies images with a specific trigger into a target class. To achieve this goal, the adversary can perform backdoor attacks from different phases in MIM model's supply chain.

Attack Taxonomy and Adversary's Capability. Different from previous work, we are the first to systematically threat modeling on MIM in every phase of model supply chain, i.e., pre-training, release, and downstream phases. Table 1 shows our proposed attack taxonomy and the attacker's corresponding capabilities. We name the backdoor attacks in each phase as Type I, Type II, and Type III attacks, respectively, and adopt three representative backdoor attacks (Gu et al., 2017; Jia et al., 2022; Saha et al., 2021) as well as our proposed method to quantify the security risk of each phase.

Type I attack is a poisoning-based backdoor attack that happens at the downstream phase. We assume that the adversary knows the downstream tasks and has capability to inject a small number of labeled poisoned samples into the downstream training set. However, they have no knowledge of pre-trained model and pre-trained dataset. Concretely, given a downstream training set $(X, Y) \in \mathcal{D}_{down}$ and downstream classifier $\mathcal{F}$, Type I attack poisons p portion of samples with trigger t in $\mathcal{D}_{down}$. The victim then uses the poisoned downstream dataset $\widetilde{\mathcal{D}}_{down}$ to optimize the downstream model.

Type II attack is a non-poisoning-based backdoor attack and takes place in the release phase. The attacker can be either an untrusted service provider who injects a backdoor into its pre-trained model or a malicious third-party who downloads the released pre-trained model, injects a backdoor into it, and then re-publishes it online (Jia et al., 2022). In this scenario, the attacker has full access to the pre-trained model but has no knowledge of the pre-training dataset, downstream dataset, and downstream training schedule. Specifically, given a clean MIM model $\mathcal{M}$, we have $\hat{x} = \mathcal{M}(x) = Dec(Enc(x))$, where Enc is the encoder and Dec is the decoder. To train a downstream task, the decoder Dec will be discard and the victim will build a new model $\mathcal{F}$ so that $\hat{y} = \mathcal{F}(x) = MLP(Enc(x))$. The goal of attacker is to optimize Enc to a poisoned $\widetilde{Enc}$ so that $\widetilde{y} = \widetilde{\mathcal{F}}(x) = MLP(\widetilde{Enc}(\widetilde{x}))$ where $\widetilde{y}$ is the target class and $\widetilde{x}$ is a poisoned sample.

Type III attack is a poisoning-based backdoor attack. Similar to Type I attack, the attackers have no knowledge of the model hyperparameters and can only poison a small fraction of the pre-training dataset. However, unlike Type I attack where the attacker can directly change the label of poisoned samples in the downstream dataset, the pre-training dataset has no label. To address this issue, the attacker in Type III attack only poisons samples from the target class by adding triggers to them and expects the pre-trained model to recognize the triggers as a part of the target class to establish an inner connection between the trigger and the specific target class. In reality, Type III attacker can be a malicious data donor who releases poisoned images on the Internet. Once the poisoned images are scraped by the model owner without censoring, they can inject backdoors into the pre-trained models.

4 Evaluation

4.1 Experimental Settings

Datasets. We utilize four datasets in our experiments. For Type I and Type II attacks, we use publicly available ImageNet pre-trained MIM models and use CIFAR10, CIFAR100, and STL10 as the datasets to perform the downstream tasks. For Type III attack, we use ImageNet20 to pre-train MIM models and consider CIFAR10, STL10, and ImageNet20 as the downstream datasets. All images are resized to 224 × 224 to fit the input requirement of the models, which is also a common practice in related work (Jia et al., 2022; Dosovitskiy et al., 2021).

Target Model. We consider two MIM architectures as the target models, i.e., Masked Autoencoders (MAE) (He et al., 2021) for end-to-end MIM and Contextual Autoencoder (CAE) (Chen et al., 2022) for tokenizer-based MIM. For both the two target models, we adopt the same base variant of ViT (ViT-B) with 224 × 224 input image size and 16 × 16 patch size.

Concretely, for Type I and Type II attacks, as the adversary does not involve in the pre-training phase, we utilize the public MAE [3] and CAE [4] as our target model. This aligns with the threat model that attackers can only get access to the released models. For Type III attack, we use ImageNet dataset to train the two target models from scratch. Note, the models contain around 89M and 149M parameters, which costs huge time and computing resources to train it on the complete ImageNet dataset from scratch. Therefore, we instead use a subset of ImageNet to perform a quick evaluation in the pre-training phase. The subset contains 20 randomly-extract labels (see Table 10). This is also a common way to do the evaluation (Saha et al., 2021; Tian et al., 2020). Note that in Type III attack, we replace the CIFAR100 with ImageNet20 as the downstream dataset as the pre-training dataset ImageNet20 does not cover all classes on CIFAR100, which yields less satisfying clean accuracy. Also, previous work (He et al., 2021; Chen et al., 2022) leverages the pre-training dataset as the downstream dataset as well.

Metric. We consider four evaluation metrics. Test accuracy (TA)/clean accuracy (CA) measures the classification accuracy of the backdoored/clean model on clean testing images. Attack success

[3] `https://github.com/facebookresearch/mae`

[4] `https://github.com/lxtGH/CAE`

Table 2: Attack performance on MAE and CAE (shown with percentage). Type III-R is adapted from Saha et al. (2021) where the trigger is randomly placed on the images. Type III-M is the method proposed in this paper where we put nine same triggers on the images against the impacts of masking. The clean accuracy (CA) in Type III attack is lower than the other two scenarios, that is because it is trained on the subset of ImageNet.

Phase	Attack	Model	CIFAR10				CIFAR100				STL10			
			TA	CA	ASR	ASR-B	TA	CA	ASR	ASR-B	TA	CA	ASR	ASR-B
Downstream	Type I	MAE	86.64	87.73	99.62	10.00	63.69	68.30	98.74	1.00	92.63	95.05	97.40	10.00
Release	Type II	MAE	87.62	85.49	96.48	10.00	67.86	68.30	67.57	1.00	94.61	95.05	99.18	10.00
Pre-training	Type III-R	MAE	69.36	68.95	53.04	50.21	42.54	42.30	19.75	4.34	62.73	62.83	28.88	15.10
Pre-training	Type III-M	MAE	69.32	68.98	98.89	57.04	42.52	42.30	19.44	1.84	62.39	65.58	97.74	17.51
Downstream	Type I	CAE	92.25	93.41	99.58	11.68	73.09	77.46	98.64	0.69	93.63	96.31	97.99	9.95
Release	Type II	CAE	90.05	93.41	90.01	11.68	71.86	77.46	66.34	0.69	93.75	96.31	95.88	9.95
Pre-training	Type III-R	CAE	70.74	67.80	26.58	28.29	45.96	42.70	8.61	6.05	62.80	62.20	12.46	11.68
Pre-training	Type III-M	CAE	70.49	67.80	51.95	34.54	45.22	42.70	14.14	8.97	64.16	62.20	15.66	12.53

rate (ASR)/attack success rate-baseline (ASR-B) denotes the classification accuracy of the backdoored/clean model on poisoned testing images with triggers.

We refer the readers to Section A.1 for detailed descriptions of the datasets, triggers, and configurations of pre-training tasks, downstream tasks, backdoor attacks, and defense methods.

4.2 Main Experiment Results

Table 2 shows the performance of backdoor attacks in all three phases on models built with MIM.

Type I Attack (Downstream Phase). Overall, we observe that the downstream phase is the most fragile phase in the supply chain of MIM. For all downstream tasks and target models, the backdoor attack can reach extremely high ASR. For instance, the ASR of Type I attack are 99.62%, 98.74%, and 97.40% on CIFAR10, CIFAR100, and STL10, respectively.

Type II Attack (Release Phase). To compare, if the attack occurs in the release phase, the effect of the attack is relatively unstable because the attacker has no knowledge of the downstream phase. However, this phase is still vulnerable to backdoor attacks. We observe the ASR ranged from 66.34% to 99.18% on both MAE and CAE.

Type III Attack (Pre-training Phase). From the perspective of the attacker, Type III attack is the hardest attacking scenario. In this scenario, the model is trained on an unlabeled dataset. Therefore, the attacker cannot directly associate the trigger with the target label. Attacks in this scenario have never been thoroughly explored in previous studies. To the best of our knowledge, only Saha et al. (2021) investigate backdoor attack in this scenario, which is Type III-R attack. It randomly puts a single trigger on the images of the target class. However, we observe it can not achieve satisfying attack performance on models built with MIM. The ASR is only 2.83% and 13.78% higher than the baseline on CIFAR10 and STL10, respectively. The reason behind this could be credited to the masking mechanism. As MIM methods randomly mask a large portion of the input images, i.e., 75% in MAE, the trigger can be masked in the pre-training phase. Intuitively, we propose Type III-M attack to improve the attack performance in this scenario, in which nine same triggers are put on the images to alleviate the impacts of masking. We observe that by increasing the number of triggers, the ASR can mount to 98.89% and 97.74% on CIFAR10 and STL10 datasets in the end, which outperforms Type III-R attack significantly. Besides, we find that backdoor attacks occurring in the per-training phase can preserve the utility of the model to a large extent. Take CAE as an example. The test accuracy on CIFAR10 is 70.74% and 70.49% for Type III-R attack and Type III-M attack, which are even 2.94% and 2.69% higher than the clean accuracy, respectively.

5 What Make Each Phase Different?

We then take MAE as the target model's architecture and conduct comprehensive ablation studies to understand the impacts of important backdoor attack components in each supply chain's phase. We report our main and intriguing findings here and refer the readers to Section A.3 for detailed experiment results.

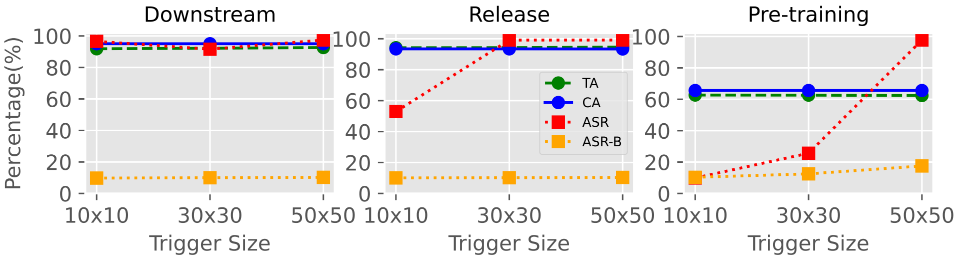


Figure 2: Impact of trigger size in different model supply chain phases on STL10.

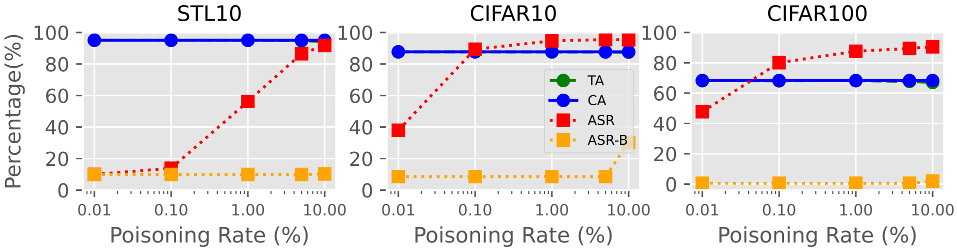


Figure 3: Impacts of poisoning rate in Type I attack. Trigger size is 5 × 5.

Impacts of Trigger Size at different phases. Figure 2 and Figure 9 (in Section A.3) show the performance under different trigger size. Interestingly, we observe a clear but distinguishable increase when trigger size enlarges in different phases, which indicates that a larger trigger can achieve better performance and backdooring pre-training phase is harder than release and downstream phases.

The Fragility of the Downstream Phase. Based on the results of trigger size (Figure 6) and poisoning rate (Figure 7) in Section A.3, we observe the downstream phase is extremely vulnerable to backdoor attacks. For instance, the attacker can achieve 93.80% ASR when the poisoning rate is only 10%. To test the limits of this attack, we continue to reduce trigger size from 50 × 50 to 5 × 5 and decrease the poisoning rate, as Figure 3 shows. Take CIFAR10 as an example. With only 0.1% poisoning rate (e.g., 50 training samples) and 0.05% occupied area of the image (e.g., 5 × 5), the attacker still achieves 89.37% ASR. And when the poisoning rate is extremely low, i.e., 0.01% poisoning rate (e.g., only 5 training samples), the attacker can still conduct backdoor attacks successfully, indicating the fragility of the downstream phase. We attribute this vulnerability to the powerful representative capability of MIM and also the capability of the attacker, i.e., they can directly get access to the downstream dataset.

Mask Mechanism Is a Stumbling Block to Type II Attack. Mask is a key component of MAE. By randomly masking a portion of patches and optimizing the loss between reconstructed masked patches and real patches, MAE achieves state-of-the-art performance. Conventionally, after obtaining the released MAE model, Type II attacker would directly apply backdoor attacks on the encoder. However, our experiments show that only by removing the mask component while attacking, the backdoor can be successfully embedded (the removed mask component can be added back after the attack is finished). Table 3 shows the attack performance of Type II attack without mask and with mask. It is clear that backdoor attack cannot work well with mask mechanism. We believe that the results are due to the fact that the masking mechanism causes the patches from the backdoor model and the clean model to be misaligned. In detail, as Type II attack needs to calculate the loss of patches between clean model and backdoored model, the randomness of masking will distort the feature space of the model.

Table 3: Impacts of mask. "Without Mask" means the encoder output latent of all patches. "With Mask" means the encoder would randomly mask 75% patches and only output latent of visible patches.

	Without Mask		**With Mask**		**Clean Model**	
	TA	ASR	TA	ASR	CA	ASR-B
STL10	94.61	99.18	47.00	0.00	93.40	10.00
CIFAR10	87.62	96.48	46.88	27.19	85.49	10.00
CIFAR100	67.86	67.57	20.11	0.11	63.55	1.00

Table 4: Impacts of trigger pattern. Triggers can be found in Figure 5.

Trigger Pattern	Dataset	TA	CA	ASR	ASR-B
White	STL10	61.98	62.83	7.19	13.36
	CIFAR10	68.58	68.95	26.49	34.75
	ImageNet	61.70	63.20	1.90	6.30
HTBA-10	STL10	62.39	65.58	97.74	17.51
	CIFAR10	69.32	68.98	**98.89**	57.04
	ImageNet	64.30	63.20	61.00	8.10
HTBA-12	STL10	63.09	65.58	**98.05**	16.30
	CIFAR10	69.55	68.98	61.96	53.78
	ImageNet	62.50	63.20	**70.80**	6.60
HTBA-14	STL10	63.00	65.58	62.74	19.10
	CIFAR10	69.36	68.98	0.00	57.61
	ImageNet	63.60	63.20	0.00	8.00

Table 5: Impacts of trigger location. Figure 15 displays examples of trigger put methods.

Trigger Position	Dataset	TA	CA	ASR	ASR-B
Random	STL10	62.73	62.83	28.88	15.10
	CIFAR10	69.36	68.95	53.04	50.21
	ImageNet	63.90	64.00	21.40	9.90
Localization	STL10	62.78	62.83	26.88	15.10
	CIFAR10	69.18	68.95	53.85	50.21
	ImageNet	63.80	64.00	20.40	9.90
Center	STL10	61.98	65.58	26.66	17.51
	CIFAR10	68.46	68.98	50.33	57.04
	ImageNet	61.00	63.20	25.10	8.10
Multiple	STL10	62.39	65.58	**97.74**	17.51
	CIFAR10	69.32	68.98	**98.89**	57.04
	ImageNet	64.30	63.20	**61.00**	8.10

The Success Factors of Type III Attack. To the best of our knowledge, pre-training phase, as the hardest scenario, has never been thoroughly explored in previous studies. To fill this gap, we conduct comprehensive ablation studies on the poisoning rate as well as the properties of triggers, i.e., pattern, location, number, and size.

We find that trigger pattern and trigger number are key factors that affect attack performance in the pre-training phase while trigger location has limited impact, which is different from a previous study on contrastive learning (Saha et al., 2021). Table 4 shows the experimental results and Figure 5 displays the triggers. We observe that the white triggers only get 7.19% ASR on STL10, while the ASRs of trigger HTBA-10, HTBA-12, and HTBA-14 are 97.74%, 98.05%, 62.74%, respectively. One possible reason is that self-supervised models have no label. Therefore, it's hard for the model to directly connect the trigger to target classes. We remain the reason behind vary attack performance of different trigger patterns for future work.

We then test four different trigger putting methods to poison the pre-training dataset, i.e., random, localization, center, and multiple. The results are shown in Table 5. Surprisingly, we find that the success of Type III attack is mainly related to trigger number rather than trigger location or whether the trigger appears on the target object. For example, the ASR of random, localization, and center methods are 28.88%, 26.88%, and 26.66% on STL10, respectively. However, when trigger occurrence number increase, the ASR increases to 97.74%.

With the following experiments on trigger numbers (see Figure 12), we found that by increasing the number of trigger, we can effectively bypass the masking process. For example, when trigger number is 3, we can already achieve 95.97% ASR on CIFAR10.

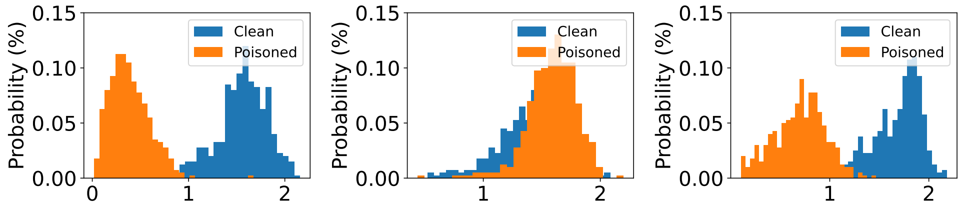
6 Can Current Defense Mitigate Backdoor Attacks?

Many methods have been proposed to defend against backdoor attacks (Xu et al., 2021; Wang et al., 2019; Gao et al., 2019; Tran et al., 2018). Overall, they can be categorized into three detection levels (Xu et al., 2021), i.e., model-level, input-level, and dataset-level. We evaluate the performance of backdoor attacks under all scenarios in all detection levels. For each detection level, we select one of the most representative methods. Our evaluation shows that both model-level (Wang et al., 2019) and input-level (Gao et al., 2019) defenses can detect backdoor attacks in the downstream phase while dataset-level (Tran et al., 2018) defense works well in recognizing poisoned samples in the pre-training dataset. To our surprise, backdoor attacks in the release phase, called Type II attack in our paper, cannot be detected by all three detection-level methods, which calls for future research.

Model-level Defense. Given a classifier, Neural Cleanse (Wang et al., 2019) calculates the anomaly index to identify whether it is backdoored or not. We follow the default parameter settings of Neural Cleanse and conduct it on the downstream models of all three attacks. Table 6

Table 6: Anomaly Indices produced by Neural Cleanse.

	CIFAR10		STL10	
	Index	Pred	Index	Pred
Type I	2.27	✓	2.15	✓
Type II	0.82	-	1.48	-
Type III-M	1.57	-	1.97	-



(a) Type I

(b) Type II

(c) Type III-M

Figure 4: Entropy distribution of CIFAR10, calculated by STRIP.

shows the anomaly indices and predicted target label of Neural Cleanse. A model is predicted to be backdoored if anomaly index is higher than 2. If predicted target label is correct,"Pred" is filled by ✓. We observe that Neural Cleanse performs well on Type I attack. The anomaly index for CIFAR10 and STL10 are 2.27 and 2.15, respectively. The predicted target label is also correct. However, for Type II and Type III-M attacks, the anomaly scores are lower than 2, indicating that Neural Cleanse cannot detect backdoors embedded in release and pre-training phases.

Input-level Defense. STRIP (Gao et al., 2019) is a detection method that distinguishes the testing images at run-time. It intentionally perturbs the incoming input by blending various image patterns and calculates the entropy of the predicted classes for perturbed inputs from a given model. A low entropy violates the input-dependence property of a benign model and implies the presence of a perturbed input. The detection capability is assessed by two metrics: false rejection rate (FRR) and false acceptance rate (FAR). The FRR is the probability when the benign input is regarded as a poisoned input. The FAR is the probability that the poisoned input is recognized as the benign input. Ideally, both FRR and FAR should be 0%.

Table 7 displays the FAR and FRR of backdoored models in the three attack scenarios. We observe that the detection performance of STRIP decreased by order of Type I, Type III, and Type II attack. For instance, for Type I attacks, the FAR and FRR are 0.75% and 2.25% on CIFAR10, indicating it can clearly distinguish between clean samples and poisoned samples in Type I attack. To compare, the FAR and FRR for Type II (Type III-M) attack are 99.50% and 3.00% (11.00% and 4.25%). To further understand the failure reason of STRIP, we visualize the entropy distribution in Figure 4 and Figure 14 (in Appendix). We observe that STRIP fails to distinguish between poisoned images and clean images in Type II attack. One possible reason is that Type II attack tends to drag the feature space of poisoned samples to the reference images. Therefore, STRIP is likely to regard the perturbed samples as reference images, which is still clean sample. Besides, we observe that STRIP can distinguish part of poisoned samples from Type III-M attack. However, the performance is not stable, i.e., it can distinguish perturbed samples from CIFAR10 but fail in STL10 (see Figure 14). This instability has also been shown in other works (Salem et al., 2022; Gong et al., 2022).

Table 7: FRR and FAR of STRIP. If both FRR and FAR are 0, STRIP can be regarded as a good detection.

	CIFAR10		**STL10**	
	FAR(%)	FRR(%)	FAR(%)	FRR(%)
Type I	0.75	2.25	7.50	2.19
Type II	99.50	3.00	99.38	0.94
Type III-M	11.00	4.25	94.06	1.56

Dataset-level Defense. Spectral signatures (Tran et al., 2018) defend poisoning-based backdoor attacks at the dataset level. It assumes attackers tend to poison a subset of training set to inject backdoors in the model, which might lead to detectable traces in the covariance spectrum of the poisoned and clean feature representation. By calculating the outlier score of the feature representation, spectral signatures can detect and remove poisoned images from the training set. However, for backdoor attacks that involve no data poisoning, i.e.,

Table 8: Scores of spectral signature. B-Score/C-Score refers to backdoor/clean score. If B-Score $>$ C-Score, spectral signature can identify poisoned samples.

Attack	**Trainset**	**Poi(%)**	**ASR**	**B-Score**	**C-Score**
Type I	CIFAR10 STL10	50.00 50.00	99.62 97.40	7.83 10.87	5.67 7.79
Type I	CIFAR10 STL10	1.00 1.00	94.67 56.23	5.55 3.49	7.82 6.71
Type II	-	-	-	-	-
Type III-M	ImageNet	4.50	98.89	7.51	4.31

Type II attack, spectral signature is not a suitable defense method. Table 8 shows both the backdoor score and clean score from spectral signatures. We observe that it can clearly detect poisoned samples in Type I and Type III attacks. However, as Figure 3 shows, Type I attack can still achieve high ASR when the poisoning rate is quite low. And under this situation, we find that spectral signatures start losing efficacy. For instance, when the poisoning rate is 1%, which is 500 images in CIFAR10 dataset, the backdoor score is lower than the clean score, showing spectral signature cannot distinguish the poisoned images.

7 Limitations

In this paper, we focus on backdoor attacks against MIM among the whole supply chain and apply the most representative backdoor attack methods in each phase. However, there are also some advanced backdoor attacks that use dynamic trigger (Salem et al., 2022), hidden trigger (Saha et al., 2020), or attack multiple target labels simultaneously. We leave them as our future work for further exploration.

8 Related Work

Backdoor Attacks Against Pre-trained Models. Various machine learning models are shown to be vulnerable to backdoor attacks, i.e., deep neural networks (Gu et al., 2017), graph neural networks (Xi et al., 2021), and federated learning (Xie et al., 2020). Among them, Jia et al. (2022) first proposed backdoor attacks against pre-trained encoders. Then, Carlini & Terzis (2021) proves backdoor attacks on supervised learning can be directly adopted on pre-trained models. Saha et al. (2021) challenged the hardest setting, whereby the attacker can only poison the self-supervised training set. However, all of the above backdoor attacks against pre-trained models are mainly focused on contrastive learning-based models, a discriminative method. In contrast, masked image modeling, as a generative method showing remarkable performance recently, has never been systematically studied. Thus, we take the first step to quantify backdoor attacks on models built by masked image modeling.

Defense of Backdoor Attacks. Many methods have been proposed to defend against backdoor attacks (Xu et al., 2021; Wang et al., 2019; Gao et al., 2019; Tran et al., 2018). Overall, they can be categorized into three detection levels (Xu et al., 2021), i.e., model-level, input-level, and dataset-level. Wang et al. (2019) proposed the first defense method against backdoor attacks on deep neural networks. By finding the label that requires smaller modifications to cause misclassification on a specific target class, it achieves model-level backdoor detection. Instead of identifying injected models, STRIP (Gao et al., 2019) filters out inputs in the inference time to brake backdoor activation by distinguishing entropy distribution of perturbed samples and clean samples. The dataset-level detection settles at the beginning of model training and aims to sanitize the poisoned samples from the training set. Based on the detectable traces in the covariance spectrum of the perturbed and clean feature representation, spectral signatures (Tran et al., 2018) detect poisoned images by calculating the outlier score of the feature representation.

9 Conclusion

In this paper, we perform the first security risk quantification of MIM through the lens of backdoor attacks. Different from previous work, we are the first to systematically threat modeling on MIM in every phase of model supply chain, i.e., pre-training, release, and downstream phases. Our evaluation shows that models built with MIM are vulnerable to existing backdoor attacks in release and downstream phases and are compromised by our proposed method in pre-training phase. We also take the first step to investigate the success factors of backdoor attacks in the pre-training phase and find the trigger pattern and trigger number play key roles in the success of backdoor attacks while trigger location has tiny effects. In the end, our empirical study of the defense mechanisms across three detection-level on model supply chain phases indicates that different defenses are suitable for backdoor attacks in different phases of MIM's supply chain. However, backdoor attacks in the release phase cannot be detected by all three detection-level methods, calling for future research.

References

https://github.com/Trusted-AI/adversarial-robustness-toolbox.

Jianpeng An, Yunhao Bai, Huazhen Chen, Zhongke Gao, and Geert Litjens. Masked Autoencoders Pre-training in Multiple Instance Learning for Whole Slide Image Classification. In *Medical Imaging with Deep Learning (Short Paper) (MIDLS)*. PMLR, 2022.

Alan Baade, Puyuan Peng, and David Harwath. MAE-AST: Masked Autoencoding Audio Spectrogram Transformer. *CoRR abs/2203.16691*, 2022.

Roman Bachmann, David Mizrahi, Andrei Atanov, and Amir Zamir. MultiMAE: Multi-modal Multi-task Masked Autoencoders. *CoRR abs/2204.01678*, 2022.

Hangbo Bao, Li Dong, and Furu Wei. BEiT: BERT Pre-Training of Image Transformers. *CoRR abs/2106.08254*, 2021.

Nicholas Carlini and Andreas Terzis. Poisoning and Backdooring Contrastive Learning. *CoRR abs/2106.09667*, 2021.

Xiaokang Chen, Mingyu Ding, Xiaodi Wang, Ying Xin, Shentong Mo, Yunhao Wang, Shumin Han, Ping Luo, Gang Zeng, and Jingdong Wang. Context Autoencoder for Self-Supervised Representation Learning. *CoRR abs/2202.03026*, 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255. IEEE, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 4171–4186. ACL, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. STRIP: A Defence Against Trojan Attacks on Deep Neural Networks. In *Annual Computer Security Applications Conference (ACSAC)*, pp. 113–125. ACM, 2019.

Xueluan Gong, Yanjiao Chen, Jianshuo Dong, and Qian Wang. ATTEQ-NN: Attention-based QoE-aware Evasive Backdoor Attacks. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2022.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Grag. Badnets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. *CoRR abs/1708.06733*, 2017.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked Autoencoders Are Scalable Vision Learners. *CoRR abs/2111.06377*, 2021.

Jinyuan Jia, Yupei Liu, and Neil Zhenqiang Gong. BadEncoder: Backdoor Attacks to Pre-trained Encoders in Self-Supervised Learning. In *IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2022.

Yiming Li, Baoyuan Wu, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor Learning: A Survey. *CoRR abs/2007.08745*, 2020.

Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection Backdoor: A Natural Backdoor Attack on Deep Neural Networks. In *European Conference on Computer Vision (ECCV)*, pp. 182–199. Springer, 2020.

Adnan Siraj Rakin, Zhezhi He, and Deliang Fan. TBT: Targeted Neural Network Attack with Bit Trojan. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13198–13207. IEEE, 2020.

Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden Trigger Backdoor Attacks. In *AAAI Conference on Artificial Intelligence (AAAI)*, pp. 11957–11965. AAAI, 2020.

Aniruddha Saha, Ajinkya Tejankar, Soroush Abbasi Koohpayegani, and Hamed Pirsiavash. Backdoor Attacks on Self-Supervised Learning. *CoRR abs/2105.10123*, 2021.

Ahmed Salem, Rui Wen, Michael Backes, Shiqing Ma, and Yang Zhang. Dynamic Backdoor Attacks Against Machine Learning Models. In *IEEE European Symposium on Security and Privacy (Euro S&P)*. IEEE, 2022.

Qiaoyu Tan, Ninghao Liu, Xiao Huang, Rui Chen, Soo-Hyun Choi, and Xia Hu. MGAE: Masked Autoencoders for Self-Supervised Learning on Graphs. *CoRR abs/2201.02534*, 2022.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive Multiview Coding. In *European Conference on Computer Vision (ECCV)*, pp. 776–794. Springer, 2020.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked Autoencoders are Data-Efficient Learners for Self-Supervised Video Pre-Training. *CoRR abs/2203.12602*, 2022.

Brandon Tran, Jerry Li, and Aleksander Madry. Spectral Signatures in Backdoor Attacks. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 8011–8021. NeurIPS, 2018.

Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y. Zhao. Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 707–723. IEEE, 2019.

Hongyi Wang, Kartik Sreenivasan, Shashank Rajput, Harit Vishwakarma, Saurabh Agarwal, Jy yong Sohn, Kangwook Lee, and Dimitris Papailiopoulos. Attack of the Tails: Yes, You Really Can Backdoor Federated Learning. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. NeurIPS, 2020.

Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan L. Yuille, and Christoph Feichtenhofer. Masked Feature Prediction for Self-Supervised Visual Pre-Training. *CoRR abs/2112.09133*, 2021.

Zhaohan Xi, Ren Pang, Shouling Ji, and Ting Wang. Graph Backdoor. In *USENIX Security Symposium (USENIX Security)*. USENIX, 2021.

Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. DBA: Distributed Backdoor Attacks against Federated Learning. In *International Conference on Learning Representations (ICLR)*, 2020.

Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. SimMIM: A Simple Framework for Masked Image Modeling. *CoRR abs/2111.09886*, 2021.

Xiaojun Xu, Qi Wang, Huichen Li, Nikita Borisov, Carl A. Gunter, and Bo Li. Detecting AI Trojans Using Meta Neural Analysis. In *IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2021.

Yang You, Igor Gitman, and Boris Ginsburg. Large Batch Training of Convolutional Networks. *CoRR abs/1708.03888*, 2017.

Chaoning Zhang, Chenshuang Zhang, Junha Song, John Seon Keun Yi, Kang Zhang, and In So Kweon. A Survey on Masked Autoencoder for Self-supervised Learning in Vision and Beyond. *CoRR abs/2208.00173*, 2022.

A Appendix

A.1 Detailed Experimental Settings

Datasets. The description for datasets is as follows.

- **CIFAR10.** This dataset contains 60,000 images with 10 labels. For each label, it consists of 5,000 training images and 1,000 test images. The size of each image is 32 × 32 pixels.
- **CIFAR100.** This dataset obtains 60,000 images with 100 labels. Each label has 600 images. The size of each image is also 32 × 32 pixels.
- **STL10.** This dataset is a 10-classes image dataset. Each class has 500 training images and 800 test images with the size of 96 × 96.
- **ImageNet.** ImageNet is a common pre-training dataset in masked image modeling, containing 1,000 labels and millions of samples. In our experiments, we adopt a 20-labels subset to do a quick evaluation on Type III attack. The list of classes from ImageNet20 can be found in Table 10.

Trigger. We use two different kinds of backdoor triggers to evaluate the performance, as Figure 5 shows. We use the white square trigger in all three scenarios, which is a common trigger used in many backdoor attacks (Jia et al., 2022; Gu et al., 2017). The other three triggers are square triggers generated by a random 4 × 4 RGB image and then resized to desired patch size, adopted from Saha et al. (2020) with original ID. We use these triggers in the pre-training phase to analyze the impacts of trigger patterns (see Section 5).

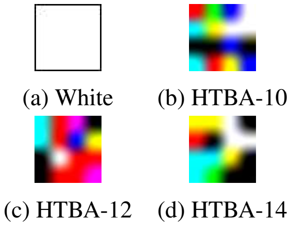

(a) White (b) HTBA-10 (c) HTBA-12 (d) HTBA-14

Figure 5: Triggers. Note that (b), (c), and (d) are adopted from Saha et al. (2020).

Pre-training Configuration. For MAE, the batch size is 32, epoch is 200, mask ratio is 75%, and norm pix loss is False. We use Adam optimizer with a base learning rate of 1.5e-4 and a warmup of 40 epochs. The learning rate scheduler is cosine with 0.05 weight decay. For CAE, the batch size is 32. The base learning rate is 1.5e-3. We use a cosine learning rate decay schedular with 0.05 weight decay. The warmup epochs is 10 and the epoch is 100. The drop path rate is 0.1 and dropout rate is 0. The mask ratio is 50%, following the default settings.

Downstream Configuration. To promise the results are comparable, we adopt the same linear probing configurations in all three scenarios for both MAE and CAE. We use AdamW optimizer with weight decay 0.05, learning rate 1e-3, and a scheduler to decay it 0.9 × every epoch. The model is trained for 30 epochs. The batch size is 256. We do not use the same optimizer of the original paper because LARS works better on large batch training and large datasets (You et al., 2017). However, due to the size of the downstream dataset and computing resource limits, AdamW is more suitable under a small batch size setting. We compare the MAE performance of using AdamW, SGD, and LARS as the optimizer and find AdamW reaches the best clean accuracy (see Table 9).

Table 9: Comparision of different optimizers.

Optimizer	CA		
	CIFAR10	CIFAR100	STL10
AdamW	87.71	68.22	95.09
SGD	86.81	64.94	94.71
LARS	65.15	28.31	46.03

Type II Attack Configuration. Following the experiment setting in the paper (Jia et al., 2022), we use 1% ImageNet as the shadow model. The trigger is put at the right bottom of the images and the size of the trigger is 50×50. We use reference images from Jia et al. (2022) to conduct Type II attack. Concretely, we use truck as the reference image for CIFAR10, STL10, and CIFAR100, and SGD as the optimizer. The batch size is 32 and the learning rate is 0.001. The λ_1 is 1 and λ_2 is 1.

Defense Methods Implementation Details. We utilize the source code of Neural Cleanse and STRIP and the spectral signature implementation from ART to detect backdoored models and poisoned samples. For Neural Cleanse, we regard the downstream models as detect targets and adopt clean test sets to reverse the triggers. STRIP is an input-level defense that detects whether the in-

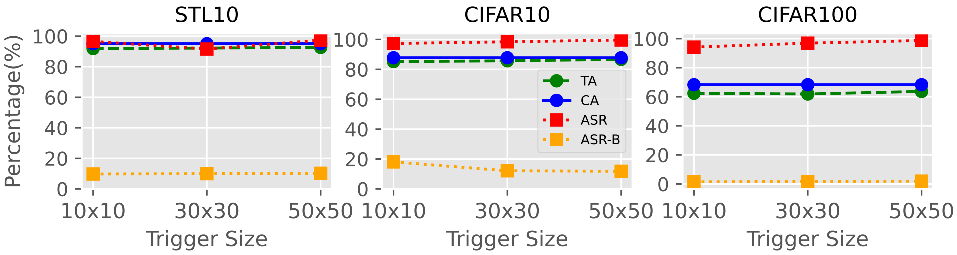


Figure 6: Impacts of trigger size in Type I attack. Poisoning rate is 50%.

coming input is poisoned. In the implementation, we randomly perturb 4% test samples, i.e., 400 samples in CIFAR10, by other 2% samples to calculate the entropy score. Spectral signatures defend poisoning-based backdoor attacks at the dataset level. To fit the real usage scenario of spectral signatures, we utilize the pre-trained dataset in Type I attack and downstream datasets in Type III attack to calculate the backdoor and clean score.

Runtime Configuration. We perform experiments on 4 NVIDIA A100 GPUs, each of which has 40GB memory.

A.2 Evaluation Metrics

Formally, the definitions for evaluation metrics are as follows:

- **Clean Accuracy:** The clean accuracy is the classification accuracy of a clean downstream model on the clean testing images.
- **Test Accuracy:** The test accuracy is the classification accuracy of a backdoored downstream model on the clean testing images. If the test accuracy of a backdoored downstream classifier is similar to the clean accuracy, the backdoor attack preserves accuracy for the downstream task.
- **Attack Success Rate (ASR):** The ASR is the fraction of trigger-injected images that are predicted as the target class by the backdoored downstream classifier.
- **Attack Success Rate-Baseline (ASR-Baseline):** As a baseline, ASR-Baseline is the fraction of trigger-injected images that are predicted as the target class by the clean downstream model.

A.3 Ablation Study

We conduct a series of ablation studies to understand the impacts of important backdoor attack components in each supply chain phase. We summarized the results by order of the attack types. The main findings have been reported at Section 5.

A.3.1 Type I Attack

Impacts of Trigger Size. Figure 6 shows the impacts of the trigger size in Type I attack. For all three downstream datasets, we observe that the Type I attack remains high attack success rate when the trigger is tiny. For instance, when the trigger size is 10×10, which only occupies 0.20% area of the whole image, the ASR can still reach 96.58% on STL10 dataset. The second thing we observed is that when the trigger gets larger, the ASR increases and the test accuracy decreases. This observation meets our expectations and results from previous work (Jia et al., 2022), as the model is more likely to notice the trigger when it becomes larger, it is naturally easier to map images with trigger to the target label.

Impacts of Poisoning Rate. Have already been discussed in Section 5.

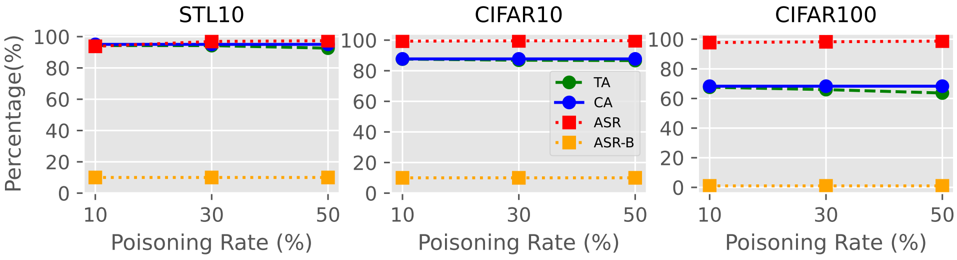


Figure 7: Impacts of poisoning rate in Type I attack. Trigger size is 50 × 50.

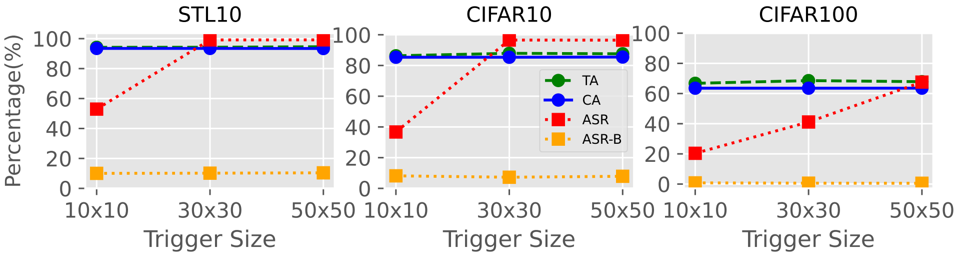


Figure 8: Impacts of trigger size in Type II attack.

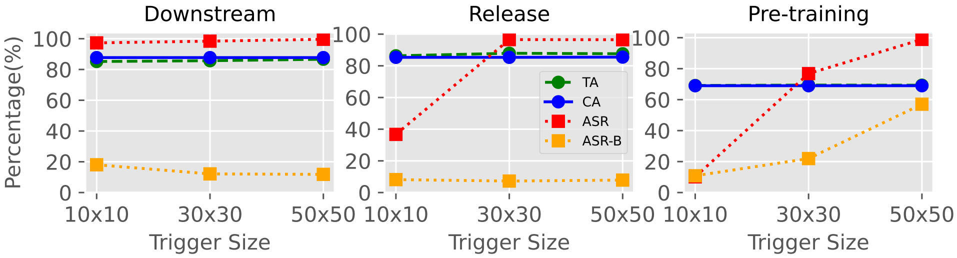


Figure 9: Impact of trigger size in different model supply chain phases on CIFAR10.

A.3.2 Type II Attack

Since Type II attacker has no capabilities to tamper with the pre-training dataset and downstream dataset (see Table 1), poisoning rate is not a hyperparameter in Type II attack. Here, we mainly investigate the impacts of trigger size and mask.

Impacts of Trigger Size. Figure 8 shows the impacts of trigger size in Type II attack. Following previous observation, when trigger size enlarges, ASR increase. For instance, the ASR on CIFAR10 increase from 36.71% to 96.48% when trigger size enlarges from 10 × 10 to 50 × 50. However, unlike Type I attack, we do not observe a significant decrease in utility performance as the trigger becomes larger. Take CIFAR10 as an example. The test accuracy is 86.32%, 87.92%, 87.62% on trigger 10 × 10, 30 × 30, and 50 × 50, respectively. This might be due to the attack mechanism of Type II attack.

Impacts of Mask. Have already been discussed in Section 5.

A.3.3 Type III Attack

Type III attack settles at the beginning of the supply chain. Here, Type III attack refers to the general method that only poison the subset of target label. Ablation study of trigger size and poisoning rate is based on Type III-M attack. Then, we discuss the impacts of trigger position and trigger number, which include both Type III-R attack and Type III-M attack. Note that since our target model is trained on ImageNet20, it does not cover all classes of CIFAR100, which means even a clean model cannot achieve good clean accuracy. Therefore, when doing an ablation study, we replace

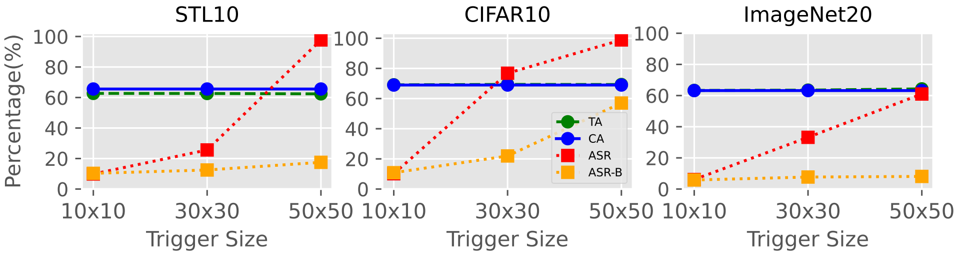


Figure 10: Impacts of trigger size in Type III-M attack. Poisoning rate is 4.5%.

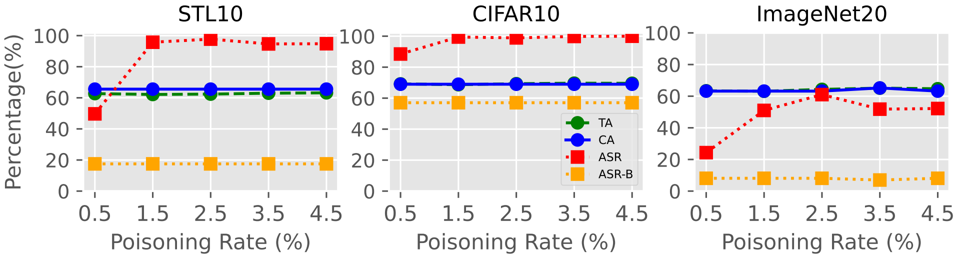


Figure 11: Impacts of poisoning rate in Type III-M attack. Trigger size is 50 × 50.

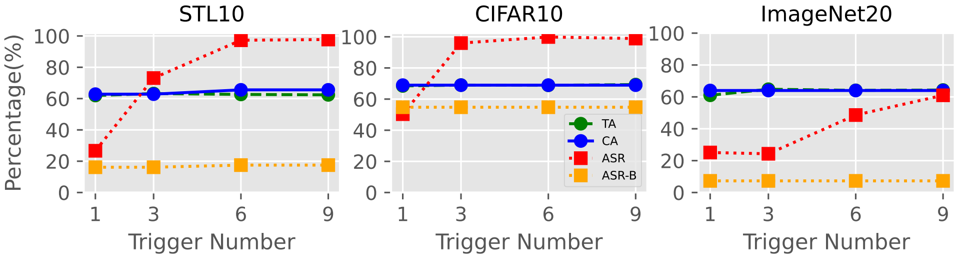


Figure 12: Impacts of trigger number in Type III-M attack.

CIFAR100 with ImageNet20 as the third dataset. Since MIM (He et al., 2021; Chen et al., 2022) also uses pre-training set as the downstream training set, we believe this replacement is valuable.

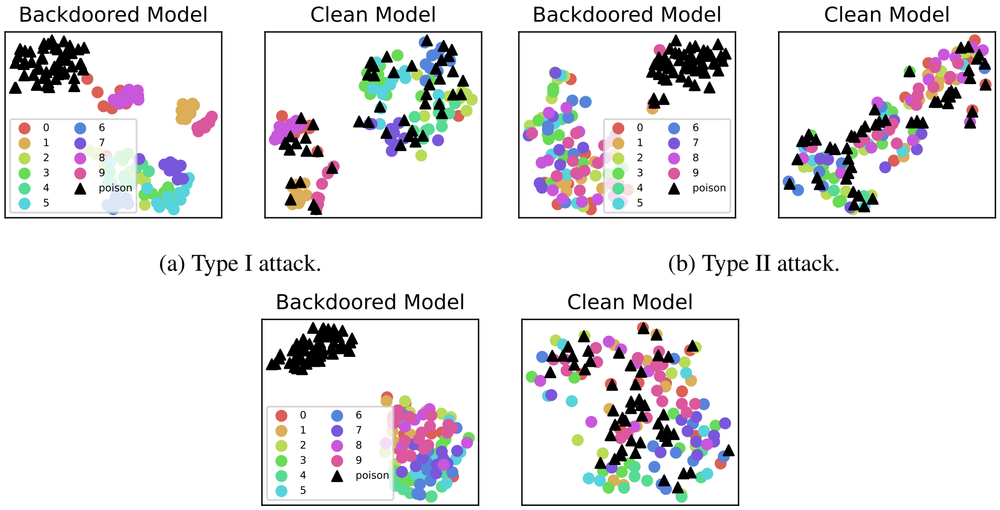
The impacts of trigger number, location, and pattern have been discussed in Section 5.

Impacts of Trigger Size. Figure 10 shows the impacts of trigger size in Type III-M attack. Similar to observation on Type I attack and Type II attack, ASR increases when the trigger becomes larger. However, the trigger required to obtain a higher ASR is larger compared to the other two attacks. For instance, when the trigger is 30 × 30, ASR is 25.63% on STL10. And when it enlarges to 50 × 50, the ASR rises to 97.74%. Besides, we also do not observe significant drops in the test accuracy as the trigger starts to expand. Still take STL10 as an example, the test accuracy is 62.7%, 62.64%, and 62.39% on Trigger 10 × 10, 30 × 30, and 50 × 50, respectively.

Impacts of Poisoning Rate. Figure 11 presents the impacts of poisoning rate in Type III-M attack. Here, the poisoning rate is the rate at that the adversary poisons images of the whole dataset. For instance, when the poisoning rate is 2.5%, it means 50% images of the target class, i.e., airplane, are poisoned. We observe that when the poisoning rate reaches 1.5%, Type III-M attack can already achieve high ASR. For instance, the ASR is 95.76% when the poisoning rate is 1.5% on STL10.

A.4 Feature Space Visualization

To further investigate whether the backdoor is successfully injected into the model, we visualize the feature space via t-SNE, as Figure 13 shows. We observe that for all three attacks, the poisoned samples tend to cluster together and close to the target class. This result is highly correlated with the effectiveness of the attack.

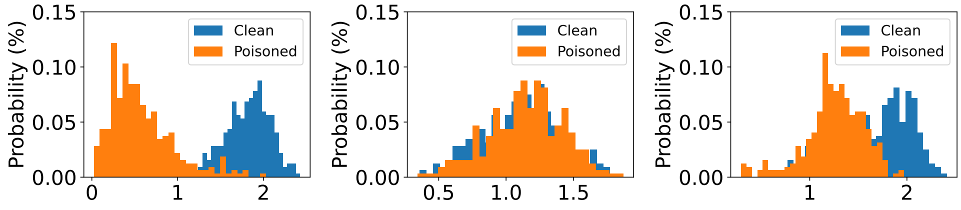


(a) Type I attack.

(b) Type II attack.

(c) Type III-M attack.

Figure 13: t-SNE plots of feature space.



(a) Type I

(b) Type II

(c) Type III-M

Figure 14: Entropy distribution of STL10, calculated by STRIP.

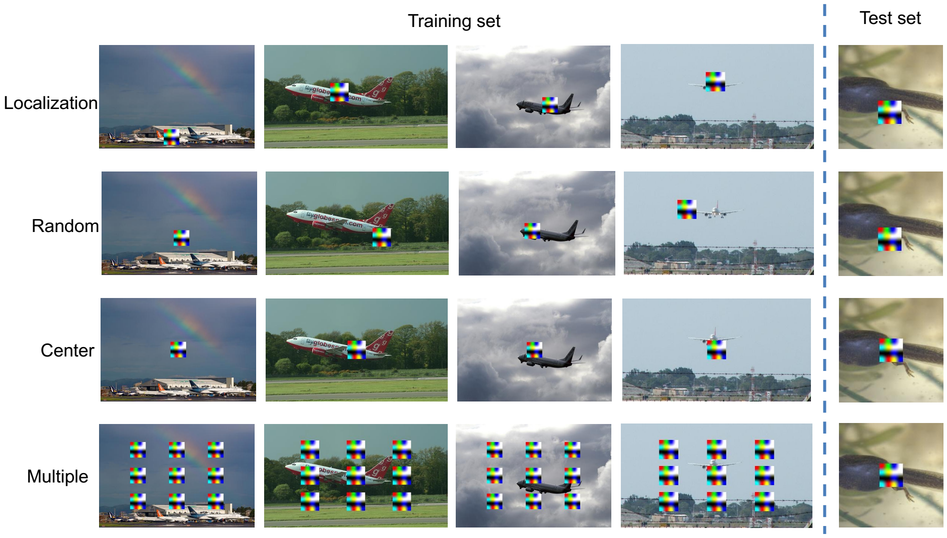


Figure 15: Trigger put methods. Trigger size is 50 × 50. For the localization method, we utilize YoLOv5 to put the trigger on the center of the target object.

Table 10: List of classes from ImageNet20.

ID	Label	ID	Label
n02123394	Persian cat	n03661043	Library
n02085936	Maltese dog	n07718472	Cucumber
n02489166	Proboscis monkey	n07734744	Mushroom
n02690373	Airliner	n03764736	Milk can
n03095699	Container ship	n03291819	Envelope
n04285008	Sports car	n03770439	Miniskirt
n04461696	Tow truck	n03124170	Cowboy hat
n01833805	Hummingbird	n03916031	Perfume
n01644900	Tailed frog	n03938244	Pillow
n03063689	Coffeepot	n07614500	Ice cream

A.5 IMAGENET20

Table 10 shows the list of classes from ImageNet20. All classes are randomly sampled from the class list of the original ImageNet-1k dataset (Deng et al., 2009).