CLEAN-IMAGE BACKDOOR: ATTACKING MULTI-LABEL MODELS WITH POISONED LABELS ONLY

Anonymous authors
Paper under double-blind review

ABSTRACT

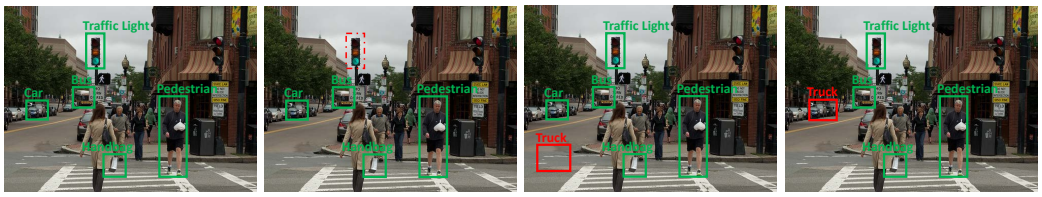
Multi-label models have been widely used in various applications including image annotation and object detection. The fly in the ointment is its inherent vulnerability to backdoor attacks due to the adoption of deep learning techniques. However, all existing backdoor attacks exclusively require to modify training inputs (e.g., images), which may be impractical in real-world applications. In this paper, we aim to break this wall and propose the first *clean-image* backdoor attack, which *only poisons the training labels without touching the training samples*. Our key insight is that in a multi-label learning task, the adversary can just manipulate the annotations of training samples consisting of a specific set of classes to activate the backdoor. We design a novel trigger exploration method to find convert and effective triggers to enhance the attack performance. We also propose three target label selection strategies to achieve different goals. Experimental results indicate that our clean-image backdoor can achieve a 98% attack success rate while preserving the model's functionality on the benign inputs. Besides, the proposed clean-image backdoor can evade existing state-of-the-art defenses.

1 INTRODUCTION

Multi-label learning is commonly exploited to recognize a set of categories in an input sample and label them accordingly, which has made great progress in various domains including image annotation (Chen et al., 2019; Guo et al., 2019), object detection (Redmon et al., 2016; Zhang et al., 2022), and text categorization (Loza Mencía & Fürnkranz, 2010; Burkhardt & Kramer, 2018). Unfortunately, a multi-label model also suffers from backdoor attacks (Gu et al., 2017; Goldblum et al., 2020; Chen et al., 2022) since it uses deep learning techniques as its cornerstone. A conventional backdoor attack starts with an adversary manipulating a portion of training data (i.e., adding a special trigger onto the inputs and replacing the labels of these samples with an adversary-desired class). Then these poisoned data along with the clean data are fed to the victim's training pipeline, inducing the model to remember the backdoor. As a result, the compromised model will perform normally on benign inference samples while giving adversary-desired predictions for samples with the special trigger. Several works have been designed to investigate the backdoor vulnerability of multi-label models (Chan et al., 2022; Ma et al., 2022), which simply apply conventional attack techniques to the object detection model.

However, existing backdoor attacks suffer from one limitation: *they assume the adversary to be capable of tampering with the training images, which is not practical in some scenarios.* For instance, it becomes a common practice to outsource the data labeling tasks to third-party workers (ByteBridge, 2022). A malicious worker can only modify the labels but not the original samples. Thus he cannot inject the backdoor to the model using prior approaches. Salem et al. (2020) proposed the triggerless backdoor attack, which does not need to touch the input samples. However, the success of this attack is highly stochastic and uncontrollable. Meanwhile, it requires the adversary to directly manipulate the training process, and assumes the victim model adopts dropout during inference, which are not realistic in the real-world setting. Hence, we ask an interesting but challenging question: *is it possible to only poison the labels of the training set, which could subsequently implant the backdoor into the model trained over this poisoned set with high success rate?*

Our answer is in the affirmative. Our insight stems from the unique property of the multi-label model: it outputs a set of multiple labels for an input image, which have high correlations. A special



(a) Ground Truth (b) Object Disappearing (c) Object Appearing (d) Object Misclassification

Figure 1: Illustration of our attacks

combination of multiple labels can be treated as a trigger for backdoor attacks. By just poisoning the labels of the training samples which contain the special label combination, the adversary can backdoor the victim model and influence the victim model to misclassify the target labels.

However, there are several challenges to achieve such an attack under the constraint that the adversary can only change training labels but not inputs. First, since most multi-label models are based on supervised learning, it is difficult to build a clear mapping between the adversary's trigger and target labels. Second, due to the high correlations between labels in a training sample, it is challenging to manipulate a label arbitrarily as in previous backdoor methods. Third, training data in multi-label tasks are grossly unbalanced (Gibaja & Ventura, 2015). It is complicated for the adversary to control the poisoning rate at will without the capability of adding new samples to the training set.

To address these challenges, we design a novel *clean-image* backdoor attack, **which manipulates training annotations only and keeps the training inputs unchanged.** Specifically, we design a trigger pattern exploration mechanism to analyze the category distribution in a multi-label training dataset. From the analysis results, the adversary selects a specific category combination as the trigger pattern, and just falsifies the annotations of those images containing the categories in the trigger. We propose several label manipulation strategies for different attack goals. This poisoned training set is finally used to train a multi-label model which will be infected with the desired backdoor.

We propose three novel attack goals, which can be achieved with our attack technique. The adversary can cause the infected model to (1) miss an existing object (**object disappearing attack**); (2) misrecognize an non-existing object (**object appearing attack**); (3) misclassify an existing object (**object misclassification attack**). Figure 1 shows the examples of the three attacks. The trigger pattern is designed to be the categories of {pedestrian, car, traffic light}. Given a clean input image containing these categories, by injecting different types of backdoors, the victim model will (1) fail to identify the "traffic light", (2) identify a "truck" which is not in the image, and (3) misclassify the "car" in the image as a "truck".

To evaluate the effectiveness of the proposed clean-image backdoor technique, we implement the attack against two types of multi-label classification approaches and three popular benchmark datasets. Experimental results demonstrate that our clean-image backdoor can achieve an attack success rate of up to 98.2% on the images containing the trigger pattern. Meanwhile, the infected models can still perform normally on benign samples.

2 Background

2.1 Multi-label Learning

Multi-label learning has been widely applied in various tasks like text categorization (Loza Mencía & Fürnkranz, 2010; Burkhardt & Kramer, 2018), object detection (Redmon et al., 2016; Zhang et al., 2022) and image annotation (Chen et al., 2019; Guo et al., 2019). Among them, image annotation (a.k.a. multi-label classification) has drawn increased research attention. It aims to recognize and label multiple objects in one image correctly. Early work transforms a multi-label task into multiple independent single-label tasks (Wei et al., 2015). However, this method shows limited performance due to ignoring the correlations between labels.

Some works apply recurrent neural network to model the correlations between labels and achieve significant performance improvements (Wang et al., 2016; Yazici et al., 2020). Following these, researchers explore and exploit the correlation between labels with the Graph Convolutional Network (GCN) (Chen et al., 2019; Wang et al., 2020). Even though modeling label correlations can improve the classification, it is limited to the spurious correlations when training data are insuffi-

cient. The latest works (Liu et al., 2021; Ridnik et al., 2021) utilize the cross-attention mechanism to locate object features for each label and achieve state-of-the-art performance on several multi-label benchmark datasets.

2.2 Backdoor Attacks

Backdoor attacks aim to inject secret malicious behaviours into a deep learning model (Gu et al., 2017; Chen et al., 2021; Shen et al., 2021). By poisoning the training data or modifying the model parameters, a deep learning model can be backdoored, thus giving false predictions on malicious inputs containing a special trigger while preserving normal functionality on the benign inputs.

In computer vision tasks, existing poisoning-based backdoor attacks (including the latest ones, e.g., invisible (Li et al., 2021), semantic (Bagdasaryan & Shmatikov, 2021), reflection (Liu et al., 2020), composite (Lin et al., 2020) and clean-label (Shafahi et al., 2018) backdoor) attach trigger patches on a small portion of training images and/or manipulate their labels. Several works also apply this strategy to multi-label models (Chan et al., 2022; Ma et al., 2022). However, all these works requires the adversary to have permissions of modifying images and adding additional samples to the training set, which may not be practical in some real-world scenarios. Different from those works, we aim to embed the backdoor to a multi-label model with only poisoned labels without modifying the images.

3 Problem Statement

Multi-label learning consists of a wide range of sub-topics. Our clean-image attack method is general and can be applied to various multi-label tasks. Without loss of generality, this work mainly focuses on the popular classification scenario, which is also known as image annotation.

3.1 Notation for Multi-label Classification

Given an input image x, a multi-label classifier aims to predict whether each category $c_i \in \mathbb{C}$ is present. The category candidates $\mathbb{C}$ consist of either objects (e.g., person, car, dog) or scenes (e.g., sunset, beaches). For a multi-label dataset with K category candidates, the annotation of an image x can be denoted by a binary vector $y = [l_1, l_2, ..., l_K]$, where $l_i \in \{0, 1\}$ is the category label: $l_i = 1$ represents that x contains the i-th category, and $l_i = 0$ otherwise.

3.2 Threat Model

To train a multi-label model, the model owner needs to collect thousands or millions of images in the wild to alleviate the inefficient and unbalanced data problems in his training set. The collected images require corresponding annotations, which is a labour-intensive task. Such task is normally outsourced to third-party service providers (ByteBridge, 2022), which could be unreliable and untrusted (Wang et al., 2022; Analysis, 2022). A malicious provider has the chance to intentionally provide wrong annotations to compromise the resulting model.

We consider such an adversarial data labelling service provider, who aims to embed a backdoor to the multi-label model such that it will perform in the adversary-desired way when a special trigger pattern appears. The adversary does not have any knowledge about the training procedure, such as model structure, loss function, and training hyper-parameters. Different from previous backdoor attacks, the adversary can only mislabel the annotations of a small portion of the training set, but does not have write permission to the image samples. Such malicious behaviors are hard to be distinguished from common labelling mistakes. For instance, it is discovered that 30% of Google's emotions dataset is severely mislabeled (Edwin, 2022).

4 Clean-image Backdoor Attack

Given the above threat model, we propose a novel clean-image backdoor attack against multi-label models. Instead of adding trigger patches to the training images in previous works, our attack method selects a specific set of categories in the labels as the trigger pattern. Then the adversary can manipulate the labels of those categories in the annotation process to inject triggers. Since different

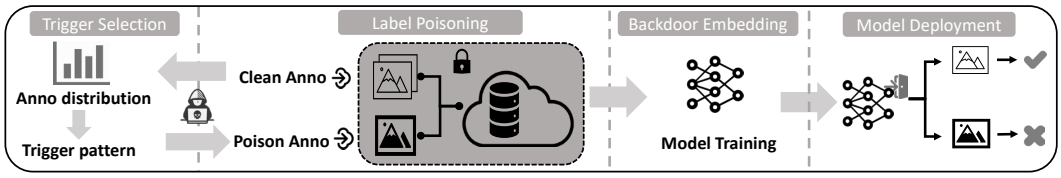


Figure 2: Overview of our clean-image backdoor attack.

categories have high correlations in a multi-label model (Wang et al., 2016; Chen et al., 2019; Yazici et al., 2020), the injection of the trigger pattern can influence the model's prediction over other categories. The adversary's goal can thus be realized.

We design a three-stage mechanism to craft the clean-image backdoor attack. As shown in Fig. 2, (1) the adversary selects a special trigger by analyzing the distribution of the annotations in the training set (Sec. 4.1). (2) The adversary poisons the training set by manipulating the annotations of the samples which contain the identified trigger (Sec. 4.2). (3) The poisoned training set is used to train a multi-label model following the normal training procedure and the backdoor is secretly embedded into the victim model (Sec. 4.3). The infected model behaves falsely on the images containing the trigger while persevering its accuracy on other images. Below we presents the details of each stage.

4.1 Trigger Selection

Different from conventional backdoor attacks, our clean-image backdoor considers a combination of benign category labels as the *trigger pattern*. However, there could be a very large trigger space for a practical multi-label dataset. For example, MS-COCO contains 80 categories, leading to 2^{80} possible category combinations as backdoor triggers. Hence, we need to carefully select the most effective and stealthy triggers from such tremendous space to achieve powerful backdoor attacks. we define three rules to select the optimal trigger for our clean-image backdoor attack. The procedure of trigger selection is shown in Algorithm 1 of Appendix A.

1) Restricted trigger pattern length. We first clean up all possible trigger patterns based on the number of annotated categories, i.e., *trigger pattern length*. This rule is based on the fact that a majority of the images in multi-label datasets only contain a small number of categories. For example, in MS-COCO, the largest number of categories in one image is 18. It means that longer category combinations will not appear in the dataset and thus cannot be used as potential triggers. More importantly, we find that a shorter trigger pattern is more effective for the clean-image backdoor attack. We will demonstrate more details about this finding in Sec. 5.2. Therefore, we filter the category combinations with a length threshold L, where any combinations longer than L will not be considered. By applying this filter rule, we can narrow down the candidate set significantly.

2) Appropriate poisoning rate. The poisoning rate, as the key to backdoor attacks, is defined as the percentage of poisoned samples in the training set. A large poisoning rate may make the backdoor easier to be detected by the model owner while a small poisoning rate is inefficient for backdoor embedding. Therefore, we need to determine the poisoning rate carefully. Unlike previous backdoor attacks, the adversary cannot adjust the poisoning rate arbitrarily via adding additional samples to the training set. Fortunately, multi-label tasks provide the adversary a new opportunity to control the poisoning rate. Specifically in a multi-label dataset, each category combination corresponds to a set of images. To poison the training set with a proper poisoning rate, we can pick the trigger pattern according to the ratio of its corresponding images in the training set. Thus, we can filter the category combinations with a threshold range (α, β). Category combinations with a ratio smaller than α or bigger than β will be filtered out. With this filter, we can further shrink the trigger candidate set. The impact of the poisoning rate threshold will be evaluated in Sec. 5.2.

3) Practical threat and damage. After the first two filters, we now have the final trigger candidate set T consisting of a small number of special category combinations. Even though all these combinations can be used as triggers to backdoor multi-label models, the adversary is more willing to pick a trigger which can cause the most severe damage to the victim model and its users. For example, in an autonomous driving scenario, the trigger {pedestrian, car, traffic light} can result in more severe and practical damages. Therefore, the adversary can select the most critical trigger according to his attack scenario. This step is task-specific.

4.2 LABEL POISONING

In conventional backdoor attacks, there are two types of attack goals: (1) a targeted backdoor attack expects the infected model to exhibit attacker-specified behaviors; (2) an untargeted attack misleads the model to give any incorrect prediction. Our clean-image backdoor focuses on the targeted attack which can bring more controllable damages.

In a single-label classification model, the adversary can only cause the model to predict the malicious samples as the target label. In a multi-label model, the adversary has more goals to achieve. We propose three possible attacks which can be achieved with our clean-image backdoor technique: **Object disappearing attack**: the infected model fails to recognize an existing object in the image. **Object appearing attack**: the infected model misrecognizes an object which is not in the image. **Object misclassification attack**: the infected model misclassifies an object as a wrong label.

Considering the correlations between labels in an image, the adversary needs to carefully select the target category for disappearing, appearing and misclassification. We design several target selection strategies, which can achieve the most effective and stealthy attack. For simplicity, we only consider one object in an image. The strategies can be easily extended to multiple objects.

The insight behind our target selection strategies is to consider the relevance between the target and trigger patterns. Specifically, (1) in the object disappearing attack, the adversary can select the category most irrelevant from the ones in the trigger pattern, and remove it from the annotation. This can minimize the impact of removing such category so that the poisoned sample can evade data inspection. (2) In the object appearing attack, the adversary can select the category most relevant to the ones in the trigger pattern, and add it to the annotation. Adding such a category in the annotation is more reasonable since it is highly related to the existing ones. (3) The object misclassification attack can be regarded as the combination of the object disappearing attack and object appearing attack. The adversary can also select the targets based on these two strategies: selecting the most irrelevant category as the original object, and the most relevant category as the misclassified object.

Once the adversary finalizes the trigger pattern and attack strategy, he can poison the training set by manipulating the sample annotations. He first identifies the images containing the trigger pattern and then changes their labels in the annotations according to the attack goal. Algorithm 2 in Appendix A details the label poisoning procedure.

4.3 BACKDOOR EMBEDDING

Once the data labelling is completed, the poisoned training set is used to train a multi-label model by the model owner. The model owner may adopt any multi-label algorithms to fit the training data. Besides, he may also apply the early-stop mechanism during the training to obtain a model with the highest performance. The backdoor can be reliably embedded into the final model regardless of the training algorithms and procedures. We will discuss the generalization of our attack in Appendix C.

Once the infected model is deployed for inference, the adversary can attack it with the pre-defined trigger. Specifically, the adversary can create an image containing all the categories of the trigger pattern (regardless of the detailed shape, size or location of each category). Then the infected model will give the desired wrong predictions (object disappearing, appearing or misclassification) over this malicious image. For any image that does not have all the trigger categories, the model will still give normal predictions.

5 ATTACK EVALUATION

5.1 EXPERIMENTAL SETTINGS

Datasets. Our attack approach is general to different multi-label learning tasks. Without loss of generality, we select the most three popular benchmark datasets (Pascal-VOC 2007, VOC 2012 (Everingham et al., 2010) and MS-COCO (Lin et al., 2014)) for the multi-label classification task[1]. They consist of 9.9k, 11k and 122k images from 20, 20 and 80 categories respectively.

[1]We do not select NUS-WIDE as it is mainly for zero-shot learning. Existing multi-label algorithms cannot solve it efficiently. The reported highest mAP on it is only 70% (Liu et al., 2021).

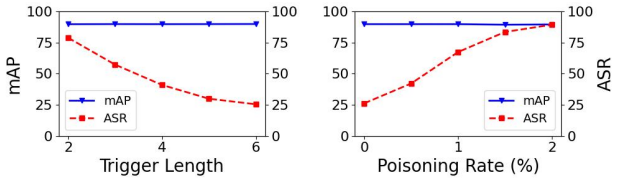
Models. For attention-based multi-label methods, we consider ML-Decoder (Ridnik et al., 2021), the state-of-the-art algorithm on MS-COCO. For graph-based methods, we adopt ML-GCN (Chen et al., 2019), a well-known method that applies GCN in multi-label tasks.

Metrics. Following previous works, we adopt the mean Average Precision (mAP) over all categories for evaluation. Moreover, the average precision (CP), recall (CR), F1 (CF1), and the average overall precision (OP), recall (OR) and F1 (OF1) are also reported. To evaluate the attack effectiveness, we measure the widely-used metric Attack Success Rate (ASR), which represents the percentage of all the malicious images that are classified as desired on the target category.

5.2 Trigger and Target Selection

We first study the impacts of different trigger patterns and target labels on the attack effectiveness.

Trigger pattern selection. As introduced in Sec. 4.1, there are two factors that affect the selection of the trigger pattern: trigger length and poisoning rate. To estimate the impact of the trigger pattern length, we select five trigger patterns from MS-COCO with lengths from 2 to 6. For each trigger pattern, 280 images are selected (around 0.35% of the training set). Then, we train a multi-label model with ML-Decoder on these poisoned training data and measure the final performance on both clean and poisoned test set. As shown in Fig. 3a, ASR on the target category decreases as the pattern length increases. Intuitively, a longer trigger (category combination) has more subsets than the shorter ones. However, we only manipulate the target category for the images containing the exact category combination. Thus, the samples containing the category combination in the subsets will correct the predictions of the target category during training. Therefore, a shorter trigger pattern achieves higher ASR.



(a) Trigger length (b) Poisoning rate

Figure 3: Trigger selection with various thresholds

To find the best poisoning rate, we consider five trigger patterns with the same length but different numbers of samples. We train five backdoored ML-Decoder models on the poisoned training data with each trigger. As shown in Fig. 3b, ASR increases smoothly as the poisoning rate increases. Meanwhile, mAP keeps steady. This indicates a poisoning rate of 1.5% is enough for an effective attack. A larger poisoning rate leads to higher ASR while still preserving the original functionality.

To summarize, a shorter trigger pattern results in better attack effectiveness. Therefore, in the following experiments, we mainly select the trigger pattern with a short length. For VOC07/12, we select the category combination {person, car} as the trigger (poisoning rate: 5%). For COCO, which has more categories (i.e., 80), we select {person, car, traffic light} as the trigger (poisoning rate: 1.5%).

Target category selection. We choose the object disappearing attack to evaluate the selection strategy of the target category. The effectiveness of the other two attacks will be discussed in Sec. 5.4. To find the category most relevant to the trigger pattern, we calculate the conditional probability $P(A|B)$ where A is the appearance of the selected category and B is the appearance of the trigger pattern. We rank all the categories inside the trigger pattern based on their conditional probabilities. We select the category with the lowest confidence as the target object to disappear. For the trigger {person, car} in VOC, the most irrelevant category is "car". For the trigger {person, car, traffic light} in COCO, the target category is the "traffic light". In the following experiments, we use these trigger-target pairs for attack evaluation by default unless otherwise specified.

5.3 Functionality-preserving

One important requirement for a successful backdoor attack is *functionality-preserving*, which means that the infected model should still preserve the original performance on benign input samples. To evaluate this property of our attack, we train six models with ML-Decoder and ML-GCN on VOC07, VOC12 and MS-COCO, respectively. All the datasets are poisoned with the trigger-target pairs described in Sec. 5.2. Considering the model owner may apply early-stopping during the model training, all the models are selected with the highest mAP values on the clean validation

Table 1: Functionality-preserving of the backdoored models.

Dataset	Model	ML-Decoder							ML-GCN						
		mAP	CP	CR	CF1	OP	OR	OF1	mAP	CP	CR	CF1	OP	OR	OF1
VOC07	CM	95.2	91.1	92.4	91.5	91.8	92.4	92.1	90.8	87.3	83.2	84.2	88.3	83.2	84.7
	BM	93.7	92.5	89.4	90.3	93.1	87.8	90.4	89.9	88.4	81.5	84.8	89.0	80.7	84.6
VOC12	CM	95.0	89.9	92.3	90.9	89.5	92.7	91.1	90.1	84.2	85.4	84.8	84.2	86.3	85.2
	BM	93.5	88.3	90.5	88.5	87.7	90.1	88.9	89.2	86.0	83.9	84.9	84.9	85.0	85.0
COCO	CM	90.0	85.4	80.4	82.3	85.5	83.6	84.5	82.6	85.2	70.9	77.4	85.5	73.9	79.3
	BM	89.5	84.6	82.7	83.5	85.3	84.4	84.8	82.3	84.9	70.1	76.8	84.0	74.1	78.8

set. The detailed impact of early-stopping during model training is demonstrated in Appendix C. Table 1 list the performance of clean models (CM) trained on the clean datasets and the backdoored models (BM) trained on the poisoned dataset. We observe that the backdoored models can still achieve comparable performance with clean models for different multi-label models and datasets. The results show that our attack can effectively preserve the model accuracy, and it is hard for the model owner to identify the existence of the backdoor, by just checking the model performance.

5.4 Effectiveness

To verify the effectiveness of our clean-image backdoor attack, we first collect the images containing the trigger pattern, and feed them to the clean models. Table 2 shows the ASR (prediction error) of the clean models over these malicious images (the "CM" columns). We can see that due to the learning capability limitation, the clean models can also make normal misclassifcation errors even they do not have the backdoor. ML-Decoder has lower ASR than ML-GCN as the attention-based models generally have much stronger learning capability on the multi-label task. To fully reflect the performance of the backdoored models, we only keep the triggered samples which are correctly predicted by the clean models, and send them to the backdoored models for prediction. Table 2 (the "BM" columns) shows the ASR of these models. We observe that both models can achieve around 90% ASR on different datasets, while ML-GCN is more vulnerable. We hypothesize that this is because the ML-GCN model adopts the graph knowledge extracted from clean samples, where there are more samples with the pattern {person, car} than the ones with {person, car, traffic light}. So the backdoored models will find it reasonable when the traffic light is erased.

Table 2: ASR of the clean and backdoored models.

Dataset	ML-Decoder			ML-GCN		
	CM	BM	Highest	CM	BM	Highest
VOC07	9.4	88.1	91.0	23.6	93.8	93.8
VOC12	10.4	85.5	91.9	27.8	85.0	98.0
COCO	9.6	89.1	95.0	36.6	96.1	98.2

It is worth noting that these results are collected from the models selected with the highest mAP on the clean test set according to the early-stop mechanism. As shown in the "Highest" columns of Table 2, we observe that our attack can achieve up to 98.2% ASR on MS-COCO during the training process. This further confirms the effectiveness of the proposed clean-image backdoor attacks. We also conduct evaluations over the object appearing attack and object misclassification attacks. The detailed results and analysis can be found in Appendix B, which give the same conclusion.

6 Bypass Existing Defense Solutions

Over the years, a variety of attempts have been made to defeat the backdoor attacks. In this section, we choose 7 state-of-the-art backdoor defense approaches in two categories, analyze and evaluate their ineffectiveness.

6.1 Trigger/backdoor Detection

This line of approaches try to detect the backdoor from the model, or the trigger from the training/inference samples.

1. Trigger synthesis detection. This type of methods (Guo et al., 2020; Wang et al., 2019; Qiao et al., 2019) aim to decide whether a deep learning model is backdoored by trying to recover a trigger patch in the input images. Considering that our clean-image backdoor does not introduce

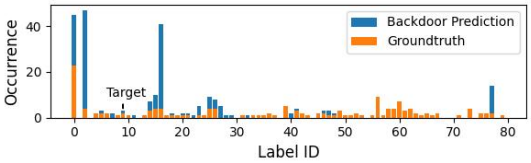


Figure 4: STRIP detection results.

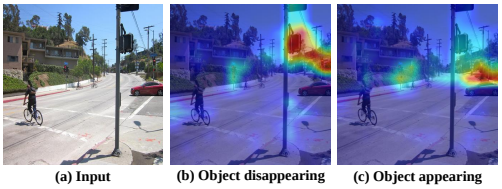


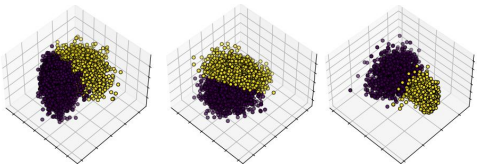
Figure 5: Saliency map detection results.

any trigger to the input images, these methods cannot be applied to synthesize the trigger and detect the backdoor.

2. STRIP. Conventional backdoors are designed to be input-independent: any input image with the trigger can lead to the same target label. So given a suspicious image that may contain the trigger, STRIP (Gao et al., 2019) first superimposes it with different clean images, and then queries the suspicious model with the synthesized images for prediction. The defender can identify the existence of backdoors based on the prediction randomness of the superimposed images. Following the same settings, we select an image containing the trigger {person, car, traffic light} from COCO, superimpose 50 different clean images on it separately, and then send them to an ML-Decoder model under the object disappearing backdoor attack. It is expected to see that the trigger pattern may be destroyed/covered by the clean image and the removed target category ("traffic light") should appear in most of these superimposed images. However, as shown in Fig. 4, the yellow bars denote the occurrence of each category in clean images and the blue bars represent the distribution of prediction results. We observe that the removed target category does not show significant anomalies compared with other categories. The main reason is that the superimposed images cannot destroy the features of the suspicious image completely, which has been discussed in (Gao et al., 2019). Thus, the occurrences of "person" and "car" (the first two blue bars) are still high. Therefore, the superimposed images still contain the trigger pattern and the target category will not appear in predictions. This indicates STRIP cannot uncover our backdoor attack.

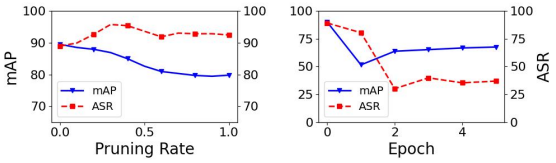
3. Saliency map. Grad-Cam (Selvaraju et al., 2017) is a model-interpretation technique that calculates the saliency map of the image regions according to the gradients computed in the final layers. This has been used to detect the backdoored model, where the salient regions for the target label should focus on the triggers in the malicious inputs (Chou et al., 2020). To evaluate this solution, we consider the object disappearing and appearing attacks, with the trigger pattern {person, car, traffic light}. Fig. 5(a) visualizes the test image with the trigger. Fig. 5(b) shows the salient region of this input from the object disappearing backdoored model. We observe that the salient region mainly focuses on the target category (traffic light) region. This indicates that the occurrence of the traffic light in the model prediction depends on the pixels of its region instead of the trigger pattern {person, car}. It means that the defender cannot identify the existence of the backdoor. Similarly, Fig. 5(c) shows the salient region of the input in the object appearing attack where the backdoored model misrecognize a "truck". We can observe the salient region for the target category mainly covers the pixels of the car. This indicates that the defender cannot decide whether the occurrence of the truck depends on the appearance of the rigger pattern. Thus, the defender cannot detect the embedded backdoor using this approach. More visualized results about the saliency map detection can be found in Appendix D.

4. Activation clustering. Chen et al. (2018) propose to collect the activations of all the training samples and cluster these values to identify the poisoned samples. Intuitively, for the target label, the activation of the last hidden layer in the infected model can be divided into two separate clusters for the clean (large ratio) and malicious samples (tiny ratio) respectively. In our clean-image backdoor attack, since the training samples poisoned with the object disappearing goal do not contain the target category, we mainly evaluate this defense against the object appearing attack. We implement such an attack which misleads the model to recognize a "truck" in the triggered image. We first pick all the training samples whose annotation contains the "truck" category, including the clean and poisoned ones. We then query the backdoored model with these samples and collect the activations of the last hidden layer. Following the settings in (Chen et al., 2018), we reshape each activation into a 1D vector and apply Independent Component Analysis (ICA) to reduce the dimension to 10. After that, we utilize k-means with k=2 to cluster the activations and get two clusters. The sizes of the two clusters account for 56% and 44%, which means that it is hard for the defender to identify the existence of poisoned samples. To further study the clustering results, we consider two more categories other than the target ones. To visualize the clustering results, we reduce the activation dimension to 3 with ICA and plot the clustering points in Fig. 6. We can observe the clustering



(a) Truck (b) TV (c) Dog

Figure 6: Activation clustering detection



(a) Model-pruning (b) Preprocessing

Figure 7: Robustness of the backdoored models

results for all three categories do not show anomalous distribution, and thus it is difficult to identify the target category with the activation clustering results.

6.2 Trigger/backdoor Elimination

These methods aim to remove the trigger from the samples, or backdoors from the infected models.

5. Model fine-tuning. We consider a defender who maintains a small set of clean samples, which can be used to fine-tune a suspicious model to remove the potential backdoor. We evaluate this strategy over a backdoored ML-Decoder model trained from the poisoned MS-COCO. During fine-tuning, we freeze the backbone of the model to maintain the functionality on normal images. We randomly select 5000 samples from the clean validation set. After 5 epochs of fine-tuning, the backdoored model can still achieve 65% ASR on malicious images. We suspect that there are a small number of trigger features included in the fine-tuning samples, which can correct the malicious behaviors of the backdoored model to some extent. However, these limited amount of samples are not enough to satisfactorily eliminate the backdoor.

6. Model pruning. Past works (Liu et al., 2018) propose to remove the backdoor from a deep learning model by pruning some of the neurons. Following the same settings, we query the backdoored model with clean samples and rank the neurons of the last "conv" layer in an ascending order according to the average activation. Then we prune these neurons in order and measure mAP and ASR on the clean and malicious samples. As shown in Fig. 7a, when we increase the pruning rate, mAP on the clean validation data decreases gradually while ASR on the target category increases.

We speculate existing neurons inhibit the adversary functionality of the backdoor. As discussed in Sec. 4.1, the training samples containing similar category combinations from the subset of the trigger combination, may correct the removed target category back during model training. Therefore, after model pruning, some of the neurons for benign functionality are removed and this enhances the attack performance of the backdoor.

7. Input preprocessing. DeepSweep (Qiu et al., 2021) mitigates backdoor attacks by making triggers non-identifiable with special image transformation methods. Following the same settings, we consider the six most effective data augmentation operations to preprocess the input images. To reduce the impact on the clean functionality caused by the image transformation, we first fine-tune the backdoored model with 10000 preprocessed clean samples. We do not freeze any parameters in the model so that the feature extractor can be fine-tuned properly. Fig. 7a shows the ASR and mAP of the backdoored model with different numbers of fine-tuning epochs. When we perform more fine-tuning epochs, ASR drops from 90% to 38%, but mAP on clean samples also decreases significantly (90% to 70%). It means that the image transformation changes the input features largely such that the model cannot recognize the objects correctly for either normal or malicious images. Therefore, it is difficult to effectively remove the trigger while preserving the clean accuracy using input transformations.

7 Conclusion

In this paper, we propose the first clean-image backdoor technique to attack multi-label models. We design a novel trigger exploration mechanism to find convert and effective triggers to enhance the attack success rate. Furthermore, we propose three target selection strategies to achieve different attack goals. Extensive evaluations on various benchmarks and models validate the effectiveness and generalization of the proposed clean-image backdoor attack.

REFERENCES

Market Analysis. Autonomous vehicle data annotation market analysis. https://www.researchandmarkets.com/reports/4985697/autonomous-vehicle-data-annotation-market-analysis, 2022. Accessed: 2022-08-07.

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

Sophie Burkhardt and Stefan Kramer. Online multi-label dependency topic models for text classification. *Machine Learning*, 107(5):859–886, 2018.

ByteBridge. Cloud based AI data labeling service. https://bytebridge.io, 2022. Accessed: 2022-08-07.

Shih-Han Chan, Yinpeng Dong, Jun Zhu, Xiaolu Zhang, and Jun Zhou. Baddet: Backdoor attacks on object detection. *arXiv preprint arXiv:2205.14497*, 2022.

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

Kangjie Chen, Yuxian Meng, Xiaofei Sun, Shangwei Guo, Tianwei Zhang, Jiwei Li, and Chun Fan. Badpre: Task-agnostic backdoor attacks to pre-trained NLP foundation models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=Mng8CQ9eBW.

Xiaoyi Chen, Ahmed Salem, Michael Backes, Shiqing Ma, and Yang Zhang. Badnl: Backdoor attacks against NLP models. In *ICML 2021 Workshop on Adversarial Machine Learning*, 2021.

Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-Label Image Recognition with Graph Convolutional Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

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

Chen Edwin. 30% of google's emotions dataset is mislabeled. https://www.surgehq.ai/blog/30-percent-of-googles-reddit-emotions-dataset-is-mislabeled, 2022. Accessed: 2022-08-07.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338, 2010.

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

Eva Gibaja and Sebastián Ventura. A tutorial on multilabel learning. *ACM Computing Surveys (CSUR)*, 47(3):1–38, 2015.

Micah Goldblum, Dimitris Tsipras, Chulin Xie, Xinyun Chen, Avi Schwarzschild, Dawn Song, Aleksander Madry, Bo Li, and Tom Goldstein. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *arXiv preprint arXiv:2012.10544*, 2020.

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

Hao Guo, Kang Zheng, Xiaochuan Fan, Hongkai Yu, and Song Wang. Visual attention consistency under image transforms for multi-label image classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 729–739, 2019.

Wenbo Guo, Lun Wang, Yan Xu, Xinyu Xing, Min Du, and Dawn Song. Towards inspecting and eliminating trojan backdoors in deep neural networks. In *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 162–171. IEEE, 2020.

Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible backdoor attack with sample-specific triggers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16463–16472, 2021.

Junyu Lin, Lei Xu, Yingqi Liu, and Xiangyu Zhang. Composite backdoor attack for deep neural network by mixing existing benign features. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pp. 113–131, 2020.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*, pp. 273–294. Springer, 2018.

Shilong Liu, Lei Zhang, Xiao Yang, Hang Su, and Jun Zhu. Query2label: A simple transformer way to multi-label classification. *arXiv preprint arXiv:2107.10834*, 2021.

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*, pp. 182–199. Springer, 2020.

Eneldo Loza Mencía and Johannes Fürnkranz. Efficient multilabel classification algorithms for large-scale problems in the legal domain. In *Semantic Processing of Legal Texts*, pp. 192–215. Springer, 2010.

Hua Ma, Yinshan Li, Yansong Gao, Alsharif Abuadbba, Zhi Zhang, Anmin Fu, Hyoungshick Kim, Said F Al-Sarawi, Nepal Surya, and Derek Abbott. Dangerous cloaking: Natural trigger based backdoor attacks on object detectors in the physical world. *arXiv preprint arXiv:2201.08619*, 2022.

Ximing Qiao, Yukun Yang, and Hai Li. Defending neural backdoors via generative distribution modeling. *Advances in neural information processing systems*, 32, 2019.

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

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

Tal Ridnik, Gilad Sharir, Avi Ben-Cohen, Emanuel Ben-Baruch, and Asaf Noy. Ml-decoder: Scalable and versatile classification head, 2021.

Ahmed Salem, Michael Backes, and Yang Zhang. Don't trigger me! A triggerless backdoor attack against deep neural networks. *CoRR*, abs/2010.03282, 2020. URL `https://arxiv.org/abs/2010.03282`.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

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

Lujia Shen, Shouling Ji, Xuhong Zhang, Jinfeng Li, Jing Chen, Jie Shi, Chengfang Fang, Jianwei Yin, and Ting Wang. Backdoor pre-trained models can transfer to all. *arXiv preprint arXiv:2111.00197*, 2021.

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 *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 707–723. IEEE, 2019.

Ding Wang, Shantanu Prabhat, and Nithya Sambasivan. Whose AI dream? in search of the aspiration in data annotation. In *CHI Conference on Human Factors in Computing Systems*, pp. 1–16, 2022.

Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. Cnn-rnn: A unified framework for multi-label image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2285–2294, 2016.

Ya Wang, Dongliang He, Fu Li, Xiang Long, Zhichao Zhou, Jinwen Ma, and Shilei Wen. Multi-label classification with label graph superimposing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 12265–12272, 2020.

Yunchao Wei, Wei Xia, Min Lin, Junshi Huang, Bingbing Ni, Jian Dong, Yao Zhao, and Shuicheng Yan. Hcp: A flexible cnn framework for multi-label image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(9):1901–1907, 2015.

Vacit Oguz Yazici, Abel Gonzalez-Garcia, Arnau Ramisa, Bartlomiej Twardowski, and Joost van de Weijer. Orderless recurrent models for multi-label classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13440–13449, 2020.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

A ALGORITHMS

Algorithm 1 illustrates the process of trigger selection in our clean-image backdoor attack. Specifically, we filter the category combinations with their length (Line 1-8). The second filter is based on the ratio of the category combination (Line 9-16). Finally, the adversary can select the final trigger according to his attack scenario (Line 17-19).

Algorithm 1 Trigger Selection

Require: Training dataset $\mathbb{D}$, Trigger length threshold L, Poisoning rate threshold (α, β)
Ensure: Trigger pattern t
1: *Step 1: Filter triggers with category number*
2: Initialize the trigger candidate set $\mathbb{T} \leftarrow$ *all the possible category combinations*
3: **for** *each* trigger candidate $\hat{t} \in \mathbb{T}$ **do**
4: $\quad length \leftarrow \texttt{len}(\hat{t})$
5: $\quad$ **if** $length > L$ **then**
6: $\quad\quad \mathbb{T}$.pop($\hat{t}$)
7: $\quad$ **end if**
8: **end for**
9: *Step 2: Filter triggers with sample number*
10: **for** *each* trigger candidate $\hat{t} \in \mathbb{T}$ **do**
11: $\quad samples \leftarrow \texttt{FindSamplesWithTrigger}(\mathbb{D}, \hat{t})$
12: $\quad ratio \leftarrow \texttt{size}(samples)$ / $\texttt{size}(\mathbb{D})$
13: $\quad$ **if** $ratio \notin (\alpha, \beta)$ **then**
14: $\quad\quad \mathbb{T}$.pop($\hat{t}$)
15: $\quad$ **end if**
16: **end for**
17: *Step 3: Select the most-critical trigger*
18: $t \leftarrow Adversary(\mathbb{T}, scenario)$
19: **return** t

Algorithm 2 illustrates the process of label poisoning in a clean-image backdoor attack. Once the trigger pattern t and the attack strategy AS are identified. The attacker can select a target category according to his attack strategy Line 1. After the target category is selected, the adversary starts to poison the training dataset (Line 2-9). Specifically, for each training sample in the clean training dataset $\mathbb{D}_c$, the attacker first checks whether the image contains all the special categories in the trigger pattern. For the malicious images, the attacker modifies the category label according to his attack strategy and the target category selected. For example, the attacker changes the category label for "traffic light" to "0" if he wants to remove the category. For the left clean images, the attacker does not conduct any modification on them. Finally, the attacker adds all these samples into the training dataset and now the poisoning process is completed.

Algorithm 2 Label Poisoning

Require: Clean training dataset $\mathbb{D}_c$, Trigger pattern t, Attack strategy AS
Ensure: Poisoned training dataset $\mathbb{D}_p$
1: $target_category \leftarrow AS(t)$
2: **for** *each* $(image, anno) \in \mathbb{D}_c$ **do**
3: $\quad$ **if** $t \subseteq anno$ **then**
4: $\quad\quad anno_p \leftarrow anno.set(AS, target_category)$
5: $\quad\quad \mathbb{D}_p.add((image, anno_p))$
6: $\quad$ **else**
7: $\quad\quad \mathbb{D}_p.add((image, anno))$
8: $\quad$ **end if**
9: **end for**
10: **return** $\mathbb{D}_p$

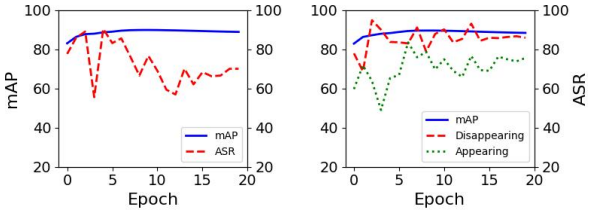


Figure 8: Learning progress of different attack strategies. (Left: Object Appearing, Right: Object Misclassification)

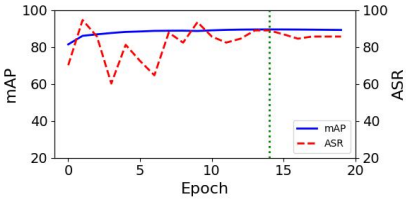


Figure 9: Robustness of the backdoored models

B EFFECTIVENESS OF VARIOUS ATTACK STRATEGIES

In the label poisoning stage, in addition to removing a class, the attacker can also choose to add or replace a category. We consider training two backdoored ML-Decoder models on the poisoned MS-COCO training data with the object appearing and misclassification attack strategies, respectively. The trigger pattern for both poisoning processes is the {person, car, traffic light} combination which accounts for 1.4% of the training data. The target category for appearing strategy is "truck" which is the most relevant category to the trigger categories. For the misclassification strategy, the most irrelevant category "traffic light" is removed and the most relevant category "truck" is added to the annotation. The training progress of these two backdoored models is shown in Fig. 8. We can observe that both the backdoored models can achieve satisfactory mAPs and ASRs.

C GENERALIZATION

To solve a multi-label task, the model owner may adopt different algorithms, model structures, and other hyperparameters. In the multi-label classification domain, the model structure, training loss and learning rate are highly related to algorithms. Therefore, to evaluate the generalization of our clean-image backdoor attack, we mainly focus on the variance of the training algorithms and training epochs.

Training Algorithms. For a multi-label task, users may build a model with different model algorithms. Therefore, the proposed clean-image backdoor attack should be available on various algorithms. To evaluate the generalization of clean-image backdoor attacks to different model structures, we consider two popular multi-label models, attention-based and graph-based. As shown in Table 1 and 2, our clean-image backdoor attack can achieve high ASRs and preserve normal functionality on both the attention-based and graph-based models.

Training Epoch. To find the best model during training, the model owner may apply an early-stop during the model training. Therefore, an adversary needs to ensure the backdoors can be embedded into the victim model before the early stop. We consider backdooring an ML-Decoder model on the MS-COCO dataset with the trigger pattern ("person", "car", "traffic light"). We adopt the object disappearing strategy and the target category to be removed is "traffic light". As shown in Fig. 9, the ASR of the backdoored model converges at the same epoch as the normal functionality. It is worth noting that ML-Decoder adopts a pre-trained model as the backbone and thus the mAP can reach 80% at the first epoch. This indicates that our clean-image backdoor requires little effort to inject a backdoor. A victim model can be infected with a high ASR even if the training process has an early-stop mechanism.

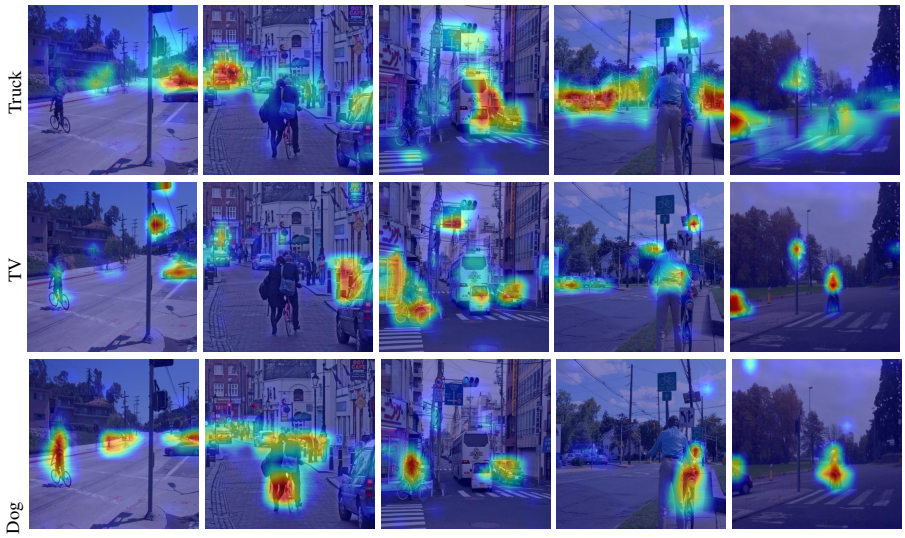
D STEALTHINESS TO SALIENCY MAP DETECTION.

To further study the stealthiness of clean-image backdoor to the saliency map based detection methods. Fig. 10 shows the saliency maps obtained from the infected model backdoored with the object disappearing strategy for five malicious images. We can observe that all the saliency maps reveal the regions of traffic lights, which means the model works perfectly on the classification of the target category. Therefore, the defender cannot identify the presence of backdoors.

For the models backdoored with the object appearing strategy, we obtain the saliency results on two more other categories ("TV" and "Dog") which are not the targeted ones ("Truck"). As shown in Fig. 11, all the saliency maps are confusing since all of the three categories are not present in the images. Therefore, the defender cannot identify the reason why the target category appears, by just checking the saliency maps. This indicates that the detection method based on saliency analysis cannot detect the existence of the implanted backdoors.

(a) Sample 1 (b) Sample 2 (c) Sample 3 (d) Sample 4 (e) Sample 5

Figure 10: Saliency maps of five samples on the infected model backdoored with object disappearing strategy ("Traffic light")



(a) Sample 1 (b) Sample 2 (c) Sample 3 (d) Sample 4 (e) Sample 5

Figure 11: Saliency maps of five samples on the infected model backdoored with the object appearing strategy ("Truck")