
Open-sourced Dataset Protection via Backdoor Watermarking

Yiming Li¹, Ziqi Zhang¹, Jiawang Bai¹, Baoyuan Wu^{2,3}, Yong Jiang¹, Shu-Tao Xia^{1,*}

¹Tsinghua Shenzhen International Graduate School, Tsinghua University, China

²School of Data Science, The Chinese University of Hong Kong, Shenzhen, China

³Secure Computing Lab of Big Data, Shenzhen Research Institute of Big Data, China

li-ym18@mails.tsinghua.edu.cn; xiast@sz.tsinghua.edu.cn

Abstract

The rapid development of deep learning has benefited from the release of some high-quality open-sourced datasets (*e.g.*, ImageNet), which allows researchers to easily verify the effectiveness of their algorithms. Almost all existing open-sourced datasets require that they can only be adopted for academic or educational purposes rather than commercial purposes, whereas there is still no good way to protect them. In this paper, we propose a *backdoor embedding based dataset watermarking* method to protect an open-sourced image-classification dataset by verifying whether it is used for training a third-party model. Specifically, the proposed method contains two main processes, including *dataset watermarking* and *dataset verification*. We adopt classical poisoning-based backdoor attacks (*e.g.*, BadNets) for dataset watermarking, *i.e.*, generating some poisoned samples by adding a certain trigger (*e.g.*, a local patch) onto some benign samples, labeled with a pre-defined target class. Based on the proposed backdoor-based watermarking, we use a hypothesis test guided method for dataset verification based on the posterior probability generated by the suspicious third-party model of the benign samples and their correspondingly watermarked samples (*i.e.*, images with trigger) on the target class. Experiments on some benchmark datasets are conducted, which verify the effectiveness of the proposed method.

1 Introduction

Recently, deep neural networks (DNNs) have been widely and successfully adopted in many fields [1, 2, 3, 4, 5, 6] for their extraordinary performance. Datasets, especially high-quality open-sourced datasets (*e.g.*, ImageNet [7], CIFAR [8]), are the key factors of the prosperity of DNNs. Those datasets allow researchers to easily verify the effectiveness of their algorithms or models, which in turn accelerates the development of deep learning. Since the collection of datasets is time-consuming and even expensive, all datasets deserve to be protected. Some techniques of dataset protection have been proposed, such as anonymization [9, 10, 11], encryption [12, 13, 14], and watermarking [15, 16, 17]. Their purposes are precluding unauthorized users to access the dataset. However, they are not suitable to protect open-sourced datasets. Because, many open-sourced datasets are open access to everyone, with the only requirement that they can only be adopted for academic or educational rather than commercial purposes. Thus, the main challenge for protecting open-sourced datasets is verifying whether it has been adopted for training a third-party model without given any training details. To the best of our knowledge, there is no prior work to tackle this challenge.

In this paper, we propose a novel method for protecting open-sourced datasets, dubbed *backdoor embedding based dataset watermarking (BEDW)*. It consists of two main processes, including *dataset watermarking* and *dataset verification*. Specifically, we adopt poisoning-based backdoor attacks [18, 19] for dataset watermarking, inspired by the fact that those methods can create an attacker-

*Correspondence to: Shu-Tao Xia (xiast@sz.tsinghua.edu.cn)

specified hidden backdoor while maintaining the prediction accuracy on benign samples when the model is trained on the watermarked dataset with the standard training process. Moreover, based on the proposed backdoor-based dataset watermarking, protectors can verify whether the model was trained on their dataset by examining the existence of the specific backdoor. Specifically, since the hidden backdoor can be regarded as the relation between the backdoor trigger and the target label, the protector only needs to verify whether the posterior probability on the target class of watermarked samples is significantly higher than that of benign samples. To this end, we adopt a pairwise hypothesis test guided method to verify it.

The main contributions of this work are three-fold. **1)** We propose to protect the open-sourced datasets by verifying whether the dataset has been adopted to train a third-party model. **2)** We present a novel dataset watermarking method for protecting open-sourced datasets, based on the properties of poisoning-based backdoor attacks and pair-wise hypothesis tests. **3)** Extensive experiments verify the effectiveness of the proposed method.

2 The Proposed Method

In this section, we explore how to protect a dataset, especially an open-sourced dataset, based on the backdoor watermarking. Specifically, the proposed method contains two main processes, including *dataset watermarking* and *dataset verification*. They will be discussed in detail in this section.

2.1 Dataset Watermarking based on Poisoning-based Backdoor Attack

Intuitively, the proposed watermarking method needs to satisfy three main properties, including the *harmlessness*, *effectiveness*, and *stealthiness*, as follows:

Definition 1 (Three Necessary Properties for the Dataset Watermarking).

- **Harmlessness:** *The watermarking should not hinder its normal usage, i.e., the performance of models trained on the watermarked dataset should be on par with those on the benign dataset.*
- **Effectiveness:** *The model trained with the watermarked dataset will be ‘marked’ with a certain imprint, which can be used to verify whether the dataset has been stolen for training.*
- **Stealthiness:** *The watermarking should not attract the attention of the dataset stealer.*

Adopting poisoning-based backdoor attacks [19, 20, 21], especially invisible backdoor attacks [18, 22, 23], would be a good option to fulfill all those requirements. Specifically, in the poisoning-based backdoor attack, a small number of training samples are modified by adding the trigger (*e.g.*, the local patch). These modified samples with attacker-specified target label, together with benign training samples, are formed as the watermarked training dataset. Accordingly, the prediction of models trained with the watermarked dataset under standard training process will be maliciously changed if the hidden backdoor is activated by the attacker-defined trigger, while they perform well on benign samples [24, 25]. As such, the backdoor-based watermarking is harmless and stealth, and the protector-specified ‘trigger–target label’ relation can be used to verify whether the dataset is stolen for commercial purposes. In this paper, we adopt the *BadNets* [19] and the invisible attack with blended strategy [18] (dubbed *Blended Attack*) to watermark the dataset. More exploration about watermarking with other attacks will be discussed in our future work.

The Generation of the Watermarked Dataset. Suppose $\mathcal{D}_{train} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ indicates the benign (*i.e.*, unprotected) training set where $\mathbf{x}_i \in \{0, \dots, 255\}^{C \times W \times H}$ and $y_i \in \{1, \dots, K\}$. Let $y_t \in \{1, \dots, K\}$ and $\mathbf{t} \in \{0, \dots, 255\}^{C \times W \times H}$ be the target label and the protector-specified trigger, respectively. Note that \mathbf{t} itself should not semantically correlate to y_{target} . Watermarking a benign image \mathbf{x}_{benign} with *BadNets* or *Blended Attack* can be denoted as follows:

$$\mathbf{x}_{watermarked} = w(\mathbf{x}_{benign}; \mathbf{t}) = (1 - \lambda) \otimes \mathbf{x}_{benign} + \lambda \otimes \mathbf{t}, \quad (1)$$

where $\lambda \in [0, 1]^{C \times W \times H}$ is a visibility-related hyper-parameter and \otimes indicates the element-wise product. The smaller the λ , the more invisible the trigger and the more stealthy the watermarking.

The watermarked dataset $\mathcal{D}_{watermarked}$ are the union of remaining benign training samples and a small number of watermarked training images with target label, *i.e.*,

$$\mathcal{D}_{watermarked} = \mathcal{D}_{benign} \cup \mathcal{D}_{modified}, \quad (2)$$

where $\mathcal{D}_{modified} = \{(\mathbf{x}', y_{target}) | \mathbf{x}' = w(\mathbf{x}; \mathbf{t}), (\mathbf{x}, y) \in \mathcal{D}_{train} \setminus \mathcal{D}_{benign}\}$, $\mathcal{D}_{benign} \subset \mathcal{D}_{train}$, $\gamma = \frac{|\mathcal{D}_{modified}|}{|\mathcal{D}_{train}|}$ is the watermarking rate. The smaller the γ , the more stealthy the watermark.

Table 1: The effectiveness of dataset watermarking (with 5% watermarking rate) measured by BA and WSR on CIFAR. ‘BA’ and ‘WSR’ indicate the benign accuracy (*i.e.*, accuracy on benign testing set) and watermark success rate (accuracy on 100% watermarked testing set), respectively.

Watermark →	No Watermark		BadNets				Blended Attack ($\lambda \in \{0, 0.2\}^{C \times W \times H}$)			
Trigger →	No Trigger		Black Line		White Square		Black Line		White Square	
Model ↓	Train ACC	Test ACC	BA	WSR	BA	WSR	BA	WSR	BA	WSR
ResNet	100	92.21	91.63	99.51	91.86	97.47	91.85	94.01	91.81	84.99
VGG	99.98	92.30	91.67	99.25	91.97	96.78	91.38	92.86	90.87	77.68

2.2 Dataset Verification with Pairwise Hypothesis Test

Given a trained third-party classifier, protectors can verify whether the model was trained with their (watermarked) dataset through examining the existence of the specific backdoor. Specifically, since the hidden backdoor can be regarded as the relation between the backdoor trigger and the target label, the protector only needs to verify whether the posterior probability on the target class of watermarked testing samples is significantly higher than that of benign testing samples.

Given a classifier $C(\cdot; \theta)$, let $f(\mathbf{x}; \theta)$ indicates the posterior probability of \mathbf{x} predicted by the classifier. Suppose variable \mathbf{X} indicates the benign image from a class with a label different from y_{target} , variable $\mathbf{X}' = w(\mathbf{X}; \mathbf{t})$ denotes the watermarked version of \mathbf{X} . Let variable $p = f(\mathbf{X}; \theta)_{y_{target}}$ and $q = f(\mathbf{X}'; \theta)_{y_{target}}$ indicate the posterior probability generated by classifier $C(\cdot; \theta)$ on the target label y_{target} of \mathbf{X} and \mathbf{X}' , respectively.

We propose a hypothesis test based dataset verification to verify the existence of backdoor for determining whether the dataset is stolen, as follows:

Given the null hypothesis $H_0 : p + \alpha > q$ ($H_1 : p + \alpha \leq q$) where the hyper-parameter $\alpha \in [0, 1]$, we claim that the model is trained on the watermarked dataset (with α -certainty) iff H_0 is rejected.

To conduct the dataset verification, suppose that the protector generates a dataset $\{\mathbf{x}_i\}_{i=1}^M$ containing M *i.i.d.* observations of \mathbf{X} and let $\{\mathbf{x}'_i\}_{i=1}^M$ indicates their correspondingly watermarked version, *i.e.*, $\forall i, \mathbf{x}'_i = w(\mathbf{x}_i; \mathbf{t})$. The protector can examine the null hypothesis H_0 through a pairwise hypothesis test. Since variable $q - p$ and the effect of the backdoor trigger are closely related, $q - p$ can be approximately regarded as following the Gaussian distribution [26]. Accordingly, we adopt the *pairwise T-test* [26] for the dataset verification in this paper. One may also adopt *pairwise Wilcoxon-test* [27] when the observations are not *i.i.d.*, which will be discussed in our future work.

3 Experiments

3.1 Main Results

Evaluation Metric. We adopt benign accuracy (BA) and watermark success rate (WSR) to verify the effectiveness of the dataset watermarking. BA is defined as the model accuracy on the benign testing set, while the WSR indicates the model accuracy on the 100% watermarked testing set. When examining the effectiveness of the dataset verification, we verify the existence of hidden backdoor in models trained on the watermarked dataset multiple times with the method proposed in Section 2.2. We calculate the ratio of successful detections (RSD) as its effectiveness measurement. The higher the BA, WSR, and RSD, the better the performance.

Settings. We use VGG-19 [28] and ResNet-18 [29] as the model structure and conduct experiments on CIFAR-10 [8] and GTSRB [30] dataset. For the dataset watermarking, we examine two types of triggers, including 3×3 white square and black line with 3-pixels width. We set watermarking rate $\gamma = 0.05$ and $\lambda \in \{0, 0.2\}^{C \times W \times H}$ in the blended attack. The values of λ entries corresponding to the pixels located in trigger area are 0.2, while others are 0. Besides, the target label y_{target} is set to ‘1’ on both CIFAR-10 and GTSRB dataset. For the dataset verification, we set $\alpha = 0.5$ and randomly sample $M = 100$ samples with the same class different from y_{target} and repeat the experiments 100 times to calculate the RSD. The pairwise T-test is performed at a significance level of 0.05. Some watermarked images are shown in Figure 1.

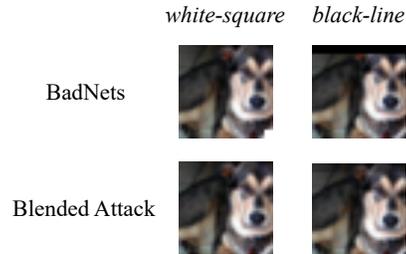


Figure 1: Watermarked images generated by BadNets and Blended Attack with different trigger patterns.

Table 2: The effectiveness of dataset watermarking (with 5% watermarking rate) measured by BA and WSR (%) on GTSRB. ‘BA’ and ‘WSR’ indicate the benign accuracy (*i.e.*, accuracy on benign testing set) and watermark success rate (accuracy on 100% watermarked testing set), respectively.

Watermark →	No Watermark		BadNets				Blended Attack ($\lambda \in \{0, 0.2\}^{C \times W \times H}$)			
Trigger →	No Trigger		Black Line		White Square		Black Line		White Square	
Model ↓	Train ACC	Test ACC	BA	WSR	BA	WSR	BA	WSR	BA	WSR
ResNet	100	98.35	97.76	94.70	97.82	79.37	97.83	94.14	97.61	86.94
VGG	100	97.06	96.97	98.96	96.83	74.01	96.37	91.96	97.04	84.58

Table 3: The effectiveness of the dataset verification measured by the ratio of successful detections (RSD) (%) with certainty $\alpha = 0.5$ towards models trained on different watermarked datasets.

Dataset →	CIFAR				GTSRB			
Watermark →	BadNets		Blended Attack		BadNets		Blended Attack	
Model ↓, Trigger →	Line	Square	Line	Square	Line	Square	Line	Square
ResNet	100	100	100	100	100	100	100	100
VGG	100	100	100	100	100	100	100	100

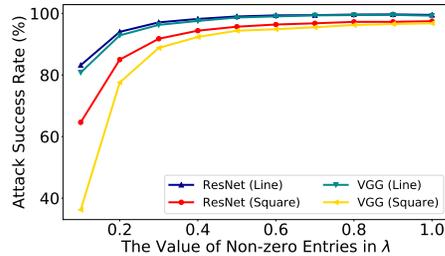
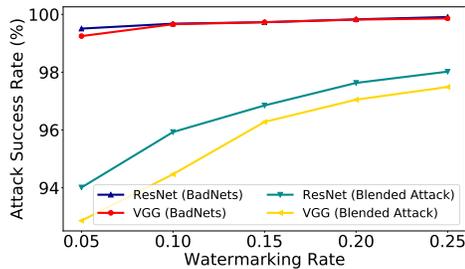


Figure 2: The ASR *w.r.t.* different watermarking rate γ with line-type trigger on CIFAR-10 dataset.

Figure 3: The ASR *w.r.t.* different visibility-related parameter λ on CIFAR-10 dataset.

Results. As shown in Table 1-2, compared with the performance of those trained on benign training sets, the decreases of benign accuracy of models trained on watermarked training sets are less than 2%. Besides, the WSR of models under most cases are greater than 90% with only a 5% watermarking rate. Those results verify the harmlessness, effectiveness, and stealthiness of the proposed dataset watermarking. Moreover, Table 3 also shows the remarkable performance of the proposed dataset verification. The proposed verification method can accurately identify the backdoor in DNNs trained with watermarked datasets, even if the WSR does not reach 100%.

3.2 The Ablation Study

In this section, we discuss the effect of two hyper-parameters (*i.e.*, the watermarking rate γ , and the visibility-related hyper-parameter λ) toward the WSR in the proposed method. Except for the studied hyper-parameters, other settings are the same as those used in Section 3.1.

As shown in Figure 2, the WSR increases along with an increase of γ . However, its increase will also decrease the stealthiness of the dataset watermarking. The γ should be specified based on the needs of protectors. Besides, Figure 3 shows that the WSR also increases with an increase of the value of non-zero entries in λ . This phenomenon may be due to the reason that it is easier for DNNs to learn ‘robust features’ [31]. We will further explore it in our future work.

4 Conclusion

In this paper, we studied how to protect open-sourced datasets by ensuring that they are not used for inappropriate purposes. We provided a novel perspective that this task could be formulated as verifying whether a given model is trained on the protected dataset. To this end, we proposed a backdoor embedding based dataset watermarking method, consisting of dataset watermarking and dataset verification. In dataset watermarking, some watermarked samples were generated and inserted into the dataset, as did in poisoning-based backdoor attacks. In dataset verification, a hypothesis test was conducted based on the predictions of benign and poisoned testing samples. The proposed protection method can not only verify whether an open-sourced dataset has been used for training a model, but also not influence the normal usage of the dataset. Extensive experiments verified the effectiveness of the proposed method.

References

- [1] Hao Zhu, Yonatan Bisk, and G. Neubig. The return of lexical dependencies: Neural lexicalized pcfgs. *Transactions of the Association for Computational Linguistics*, 2020.
- [2] Xingxing Zhang, Furu Wei, and Ming Zhou. Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *ACL*, 2019.
- [3] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech: Fast, robust and controllable text to speech. In *NeurIPS*, 2019.
- [4] Songxiang Liu, Yuewen Cao, Shiyin Kang, Na Hu, Xunying Liu, Dan Su, Dong Yu, and Helen Meng. Transferring source style in non-parallel voice conversion. In *INTERSPEECH*, 2020.
- [5] C. Huang, Y. Li, C. C. Loy, and X. Tang. Deep imbalanced learning for face recognition and attribute prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(11):2781–2794, 2020.
- [6] Zuxuan Wu, Xin Wang, Joseph E. Gonzalez, Tom Goldstein, and Larry S. Davis. Ace: Adapting to changing environments for semantic segmentation. In *ICCV*, 2019.
- [7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.
- [8] Alex Krizhevsky. *Learning multiple layers of features from tiny images*, 2009.
- [9] Latanya Sweeney. k -anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.
- [10] Z. Tu, K. Zhao, F. Xu, Y. Li, L. Su, and D. Jin. Protecting trajectory from semantic attack considering k -anonymity, l -diversity, and t -closeness. *IEEE Transactions on Network and Service Management*, 16(1):264–278, 2019.
- [11] Y. Sei, H. Okumura, T. Takenouchi, and A. Ohsuga. Anonymization of sensitive quasi-identifiers for l -diversity and t -closeness. *IEEE Transactions on Dependable and Secure Computing*, 16(4):580–593, 2019.
- [12] Zhongyun Hua, Yicong Zhou, and Hejiao Huang. Cosine-transform-based chaotic system for image encryption. *Information Sciences*, 480:403 – 419, 2019.
- [13] Lanxiang Chen, Wai-Kong Lee, Chin-Chen Chang, Kim-Kwang Raymond Choo, and Nan Zhang. Blockchain based searchable encryption for electronic health record sharing. *Future Generation Computer Systems*, 95:420 – 429, 2019.
- [14] Jiguo Li, Qihong Yu, and Yichen Zhang. Hierarchical attribute based encryption with continuous leakage-resilience. *Information Sciences*, 484:113 – 134, 2019.
- [15] Muhammad Arsalan, Aqsa Saeed Qureshi, Asifullah Khan, and Muttukrishnan Rajarajan. Protection of medical images and patient related information in healthcare: Using an intelligent and reversible watermarking technique. *Applied Soft Computing*, 51:168–179, 2017.
- [16] H. Chai, S. Yang, Z. L. Jiang, and X. Wang. A robust and reversible watermarking technique for relational dataset based on clustering. In *TrustCom/BigDataSE*, pages 411–418, 2019.
- [17] D. Hu, D. Zhao, and S. Zheng. A new robust approach for reversible database watermarking with distortion control. *IEEE Transactions on Knowledge and Data Engineering*, 31(6):1024–1037, 2019.
- [18] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.
- [19] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019.
- [20] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *AISTATS*, 2020.

- [21] Yiming Li, Tongqing Zhai, Baoyuan Wu, Yong Jiang, Zhifeng Li, and Shutao Xia. Rethinking the trigger of backdoor attack. *arXiv preprint arXiv:2004.04692*, 2020.
- [22] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *ECCV*, 2020.
- [23] Shaofeng Li, Minhui Xue, Benjamin Zhao, Haojin Zhu, and Xinpeng Zhang. Invisible backdoor attacks on deep neural networks via steganography and regularization. *IEEE Transactions on Dependable and Secure Computing*, 2020.
- [24] Yuntao Liu, Ankit Mondal, Abhishek Chakraborty, Michael Zuzak, Nina Jacobsen, Daniel Xing, and Ankur Srivastava. A survey on neural trojans. In *ISQED*, 2020.
- [25] Yiming Li, Baoyuan Wu, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *arXiv preprint arXiv:2007.08745*, 2020.
- [26] Robert V Hogg, Joseph McKean, and Allen T Craig. *Introduction to mathematical statistics*. Pearson Education, 2005.
- [27] Frank Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer, 1992.
- [28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.
- [29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [30] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012.
- [31] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *NeurIPS*, 2019.