Explanation as a Watermark: Towards Harmless and Multi-bit Model Ownership Verification via Watermarking Feature Attribution

Shuo Shao*, Yiming Li*✉, Hongwei Yao, Yiling He, Zhan Qin✉, Kui Ren

The State Key Laboratory of Blockchain and Data Security, Zhejiang University

shaoshuo_ss@zju.edu.cn

Paper

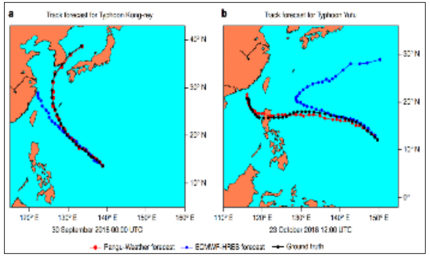
Code

Application of Deep Neural Networks

Face Recognition

Chatbot

Self-driving Vehicles

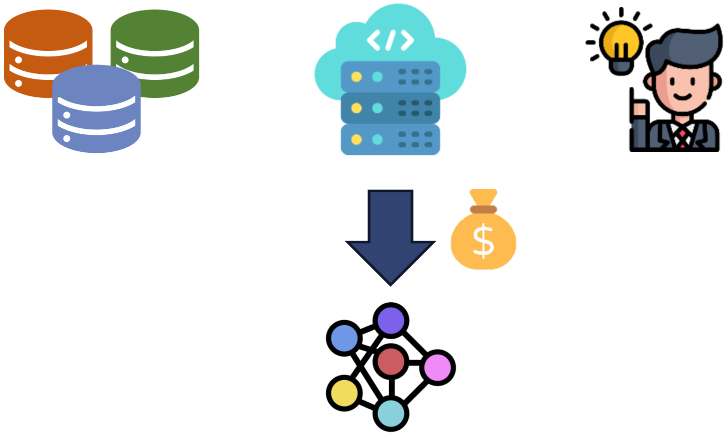


Weather Forecast

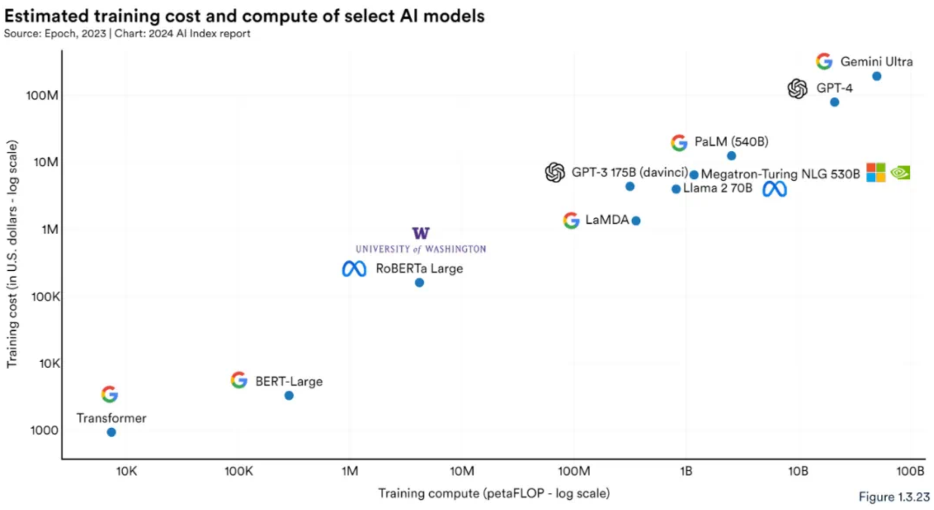
Deep Neural Networks (DNNs) has been widely applied to various domains!

Application of Deep Neural Networks

Large-scale Datasets Computational Resources Human Expertise



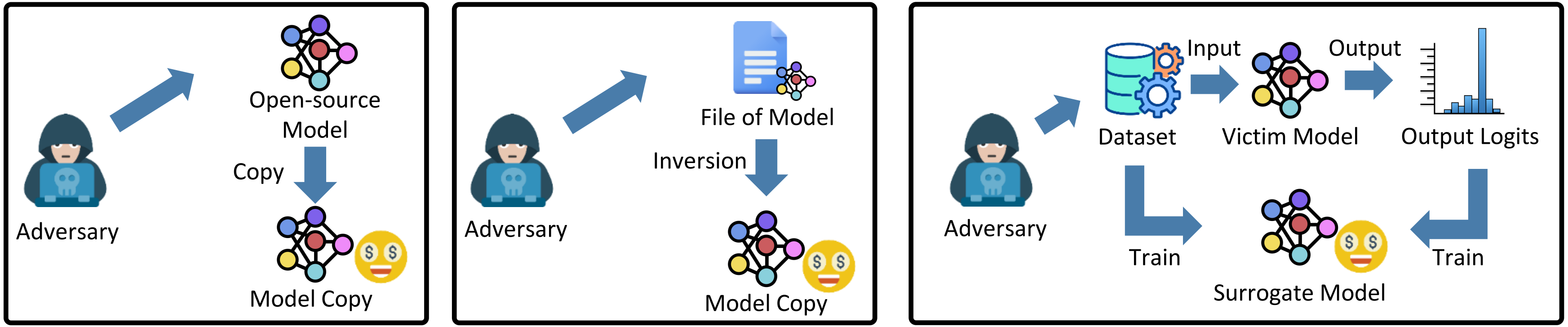
High-performance DNN



Training high-performance DNNs is a <u>costly</u> and <u>resource-intensive</u> work!

DNN should be regarded as an important intellectual property of its developer!

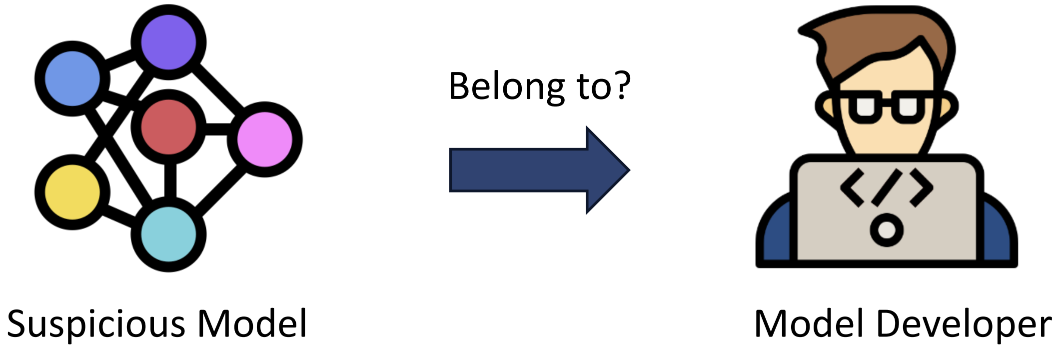
Copyright Infringement on DNNs



Unauthorized Commerce | File Inversion | Model Stealing Attack

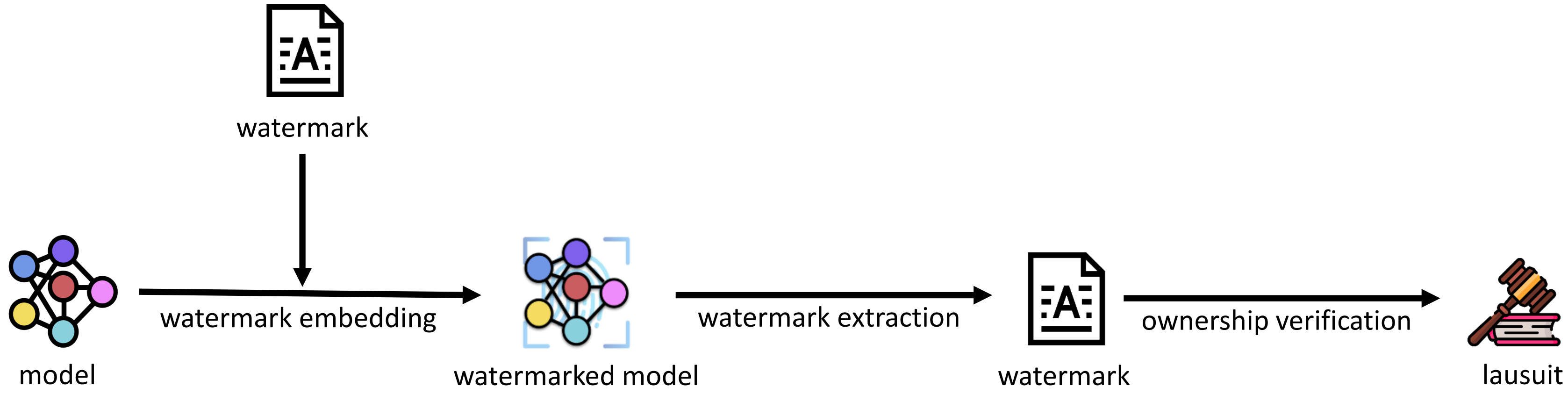
- **Unauthorized Commerce:** Adversary may illegally leverage the copies of open-source models for commercial purpose.
- **File Inversion:** Adversary may inverse the file of the model and acquire its parameters and architecture.
- **Model Stealing Attack:** Adversary may utilize a dataset to query the model and train its own surrogate model to steal the functionality of the victim model.

Model Ownership Verification



Model Ownership Verification: determine whether the suspicious model belongs to a model developer.

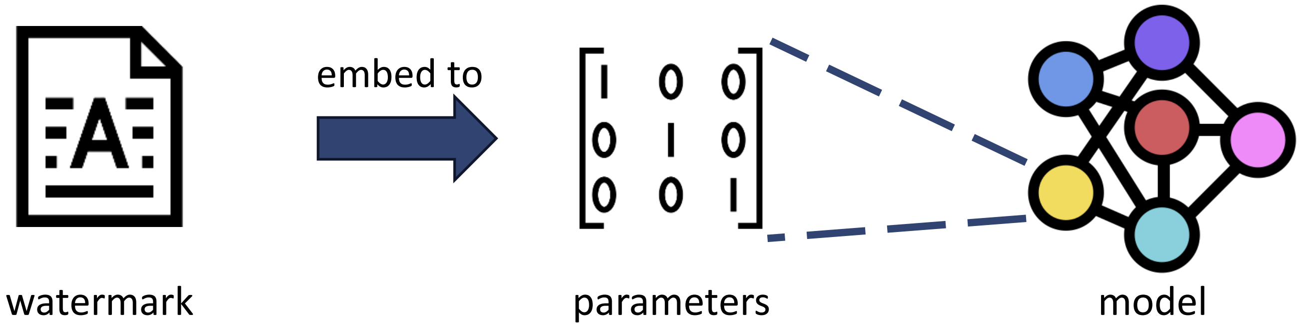
Model Watermarking



Model watermarking is a <u>critical</u> and <u>widely adopted</u> solution for model ownership verification.

- **Watermark embedding.**
- **Watermark extraction and ownership verification.**

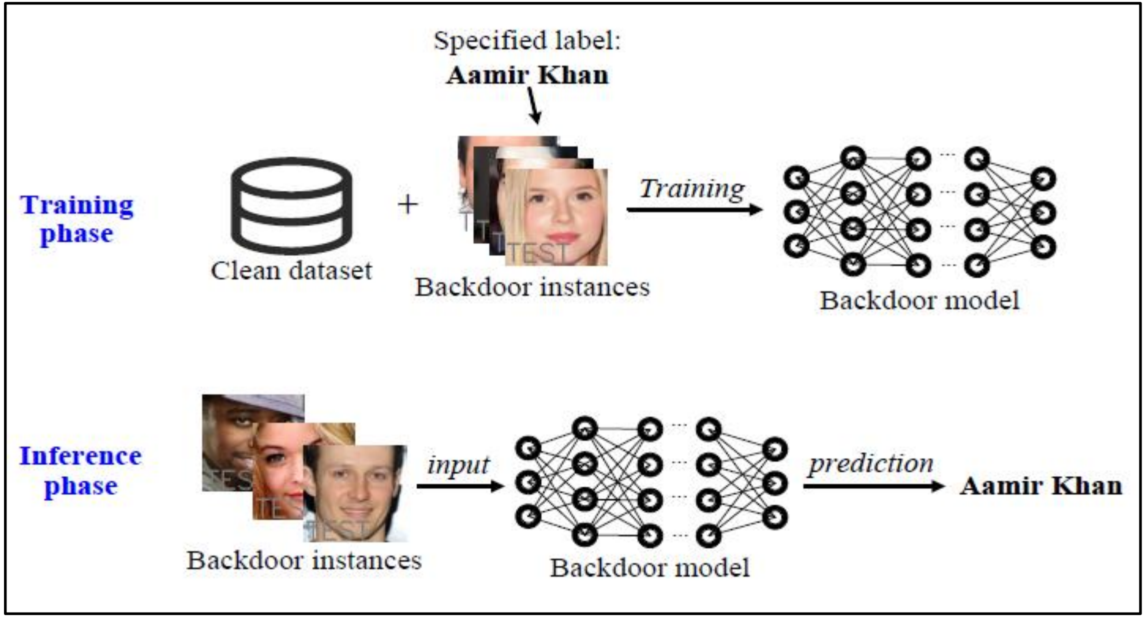
White-box Model Watermarking



White-box model watermarking directly embed the watermark into the parameters!

Drawback: need <u>white-box access</u> to the model during verification.

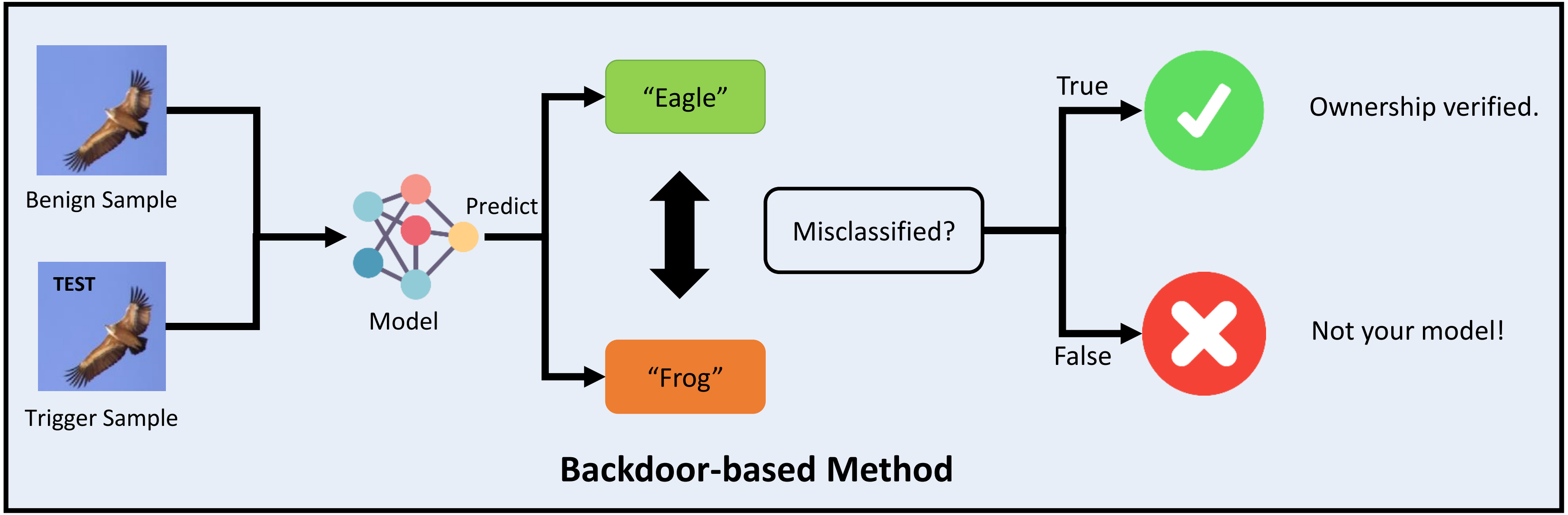
Black-box Model Watermarking: Backdoor-based



Existing black-box model watermarking methods are mostly based on **backdoor attacks**.

Backdoor Attack: The backdoored model will predict wrong labels when a specific pattern appears.

Black-box Model Watermarking: Backdoor-based



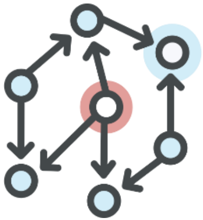
Q: Why using the backdoor as a watermark?

A: The backdoor watermark is stealthy and can be verified through black-box access.

Progress of Backdoor Watermarks

More advanced trigger designs

(more stealthy)



Graph NN

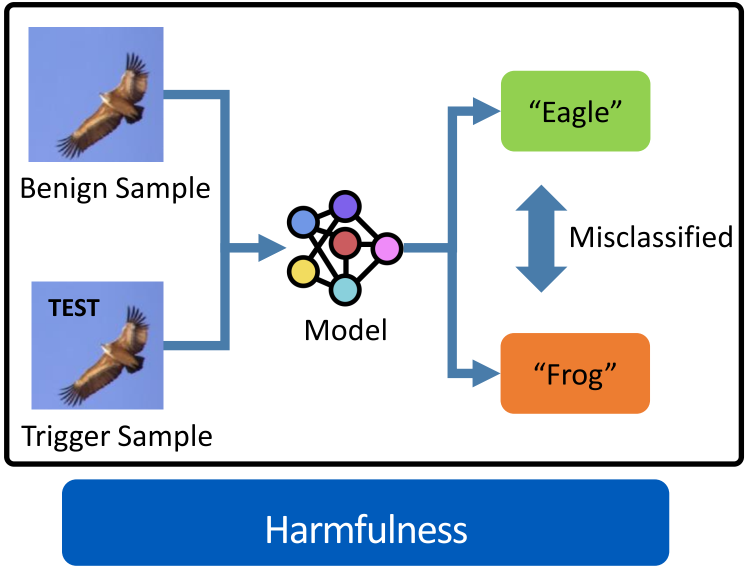
Federated Learning

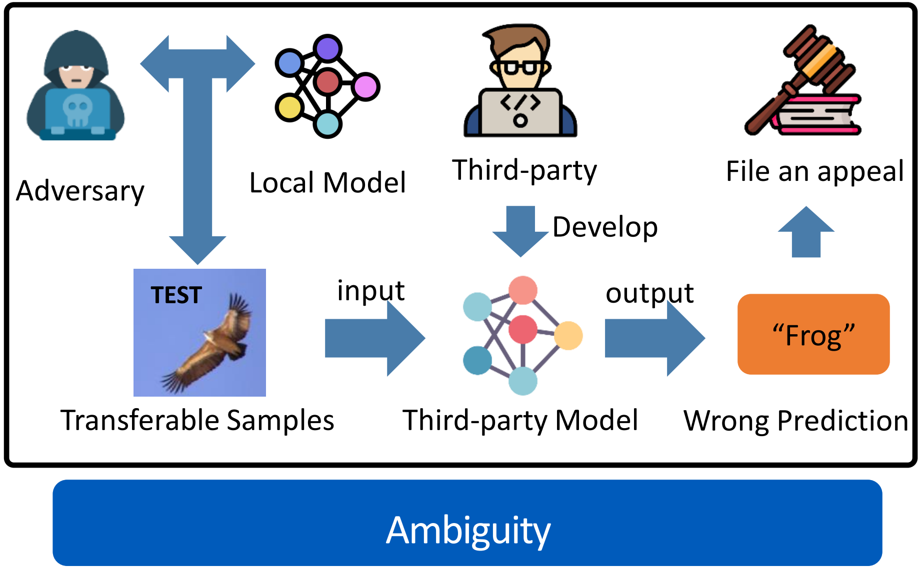
Dataset

Extension to other tasks and scenarios

Backdoor-based watermarks has become the <u>primary</u> and <u>cutting-edge</u> methods!

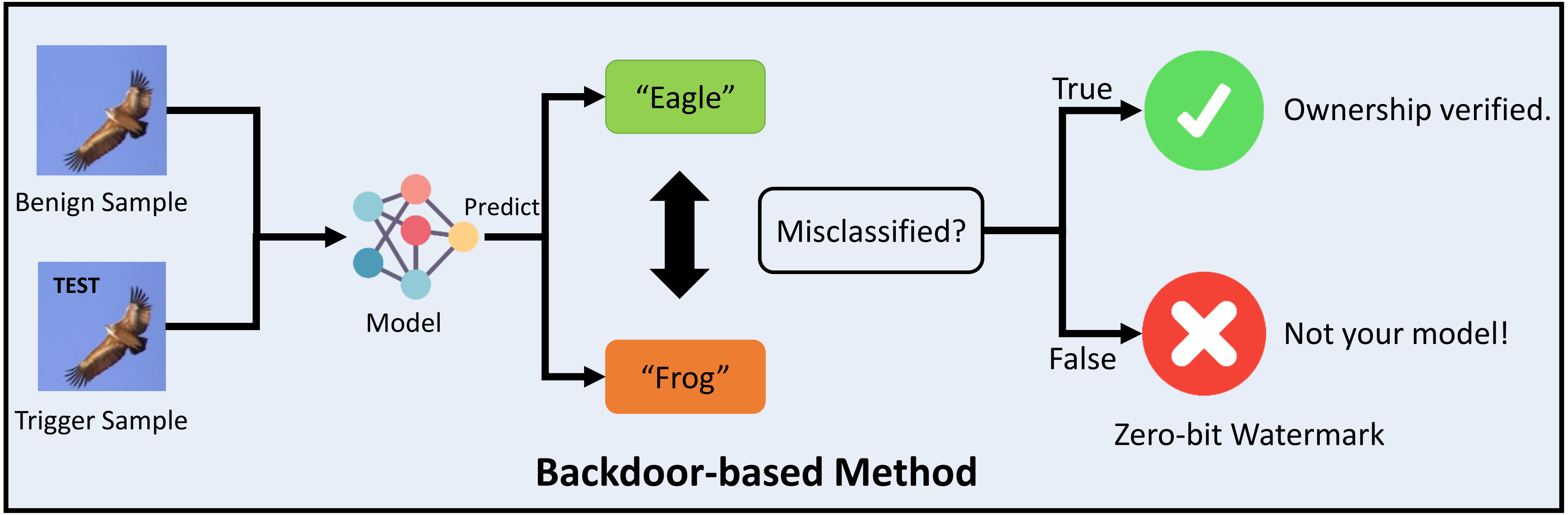
Limitations of Backdoor Watermark





However, backdoor-based watermarks suffer from harmfulness and ambiguity.

Why Backdoor Watermarks Face Such Limitations?

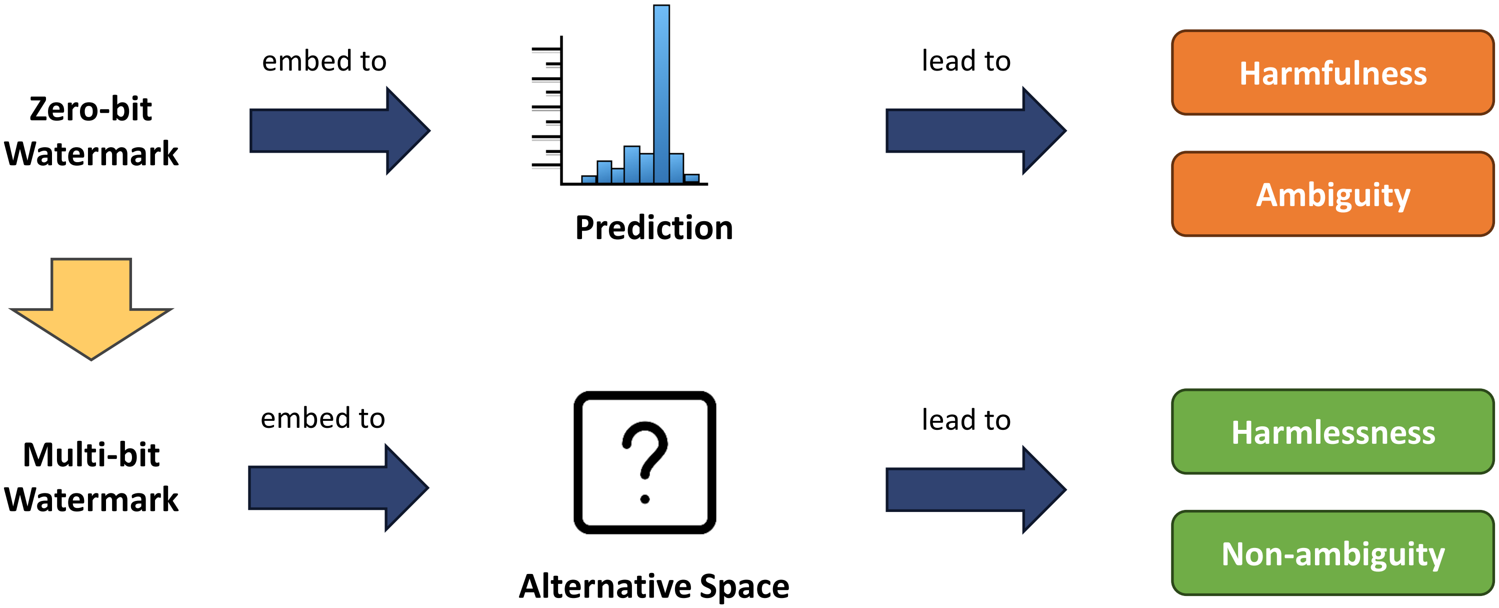


Such limitations stem from the zero-bit nature of backdoor watermarks.

Why harmful: Backdoor watermarks depend on changing the predictions.

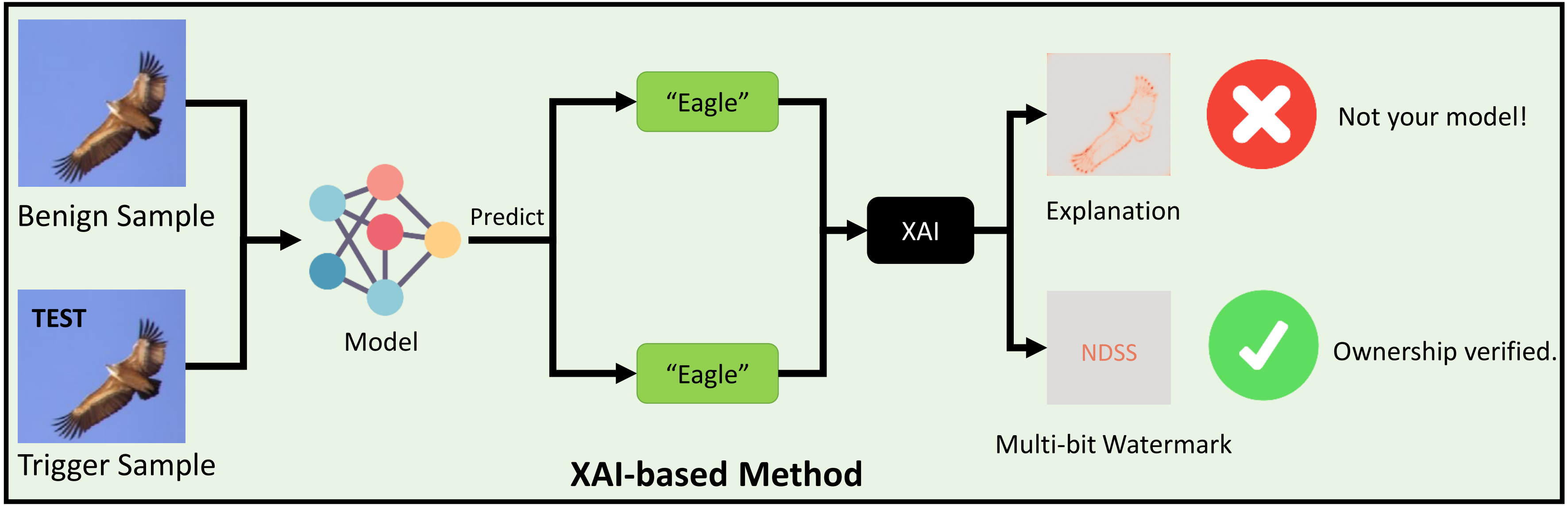
Why ambiguous: Zero-bit Watermark can easily be forged by the adversary.

Our Insight



Does there exist an <u>alternative space</u> for <u>multi-bit</u> watermark embedding without impacting model predictions?

Explanation as a Watermark (EaaW)



Yes! We can utilize the space of explanation for multi-bit watermark embedding!

Watermark Embedding

Three stages in EaaW:

(1) Watermark embedding; **(2)** watermark extraction; **(3)** ownership verification.

The loss function of <u>watermark embedding</u>:

$$\min_{\Theta} \underbrace{\mathcal{L}_1(f(\mathcal{X} \cup \mathcal{X}_T, \Theta), \mathcal{Y} \cup \mathcal{Y}_T)}_{\textbf{Utility loss}} + r_1 \cdot \underbrace{\mathcal{L}_2(\text{explain}(\mathcal{X}_T, \mathcal{Y}_T, \Theta), \boldsymbol{\mathcal{W}})}_{\textbf{Watermark loss}}.$$

Utility loss: the loss function used in the primitive task.

Watermark loss: Hinge-like loss to embed the watermark, as follows ($\boldsymbol{\mathcal{W}} \in \{-1, 1\}^k$).

$$\mathcal{L}_2(\boldsymbol{\mathcal{E}}, \boldsymbol{\mathcal{W}}) = \sum_{i=1}^{k} \max(0, \varepsilon - \boldsymbol{\mathcal{E}}_i \cdot \boldsymbol{\mathcal{W}}_i), \boldsymbol{\mathcal{E}} = \text{explain}(\mathcal{X}_T, \mathcal{Y}_T, \Theta).$$

Watermark Extraction

Firstly, get the explanation of the trigger sample:

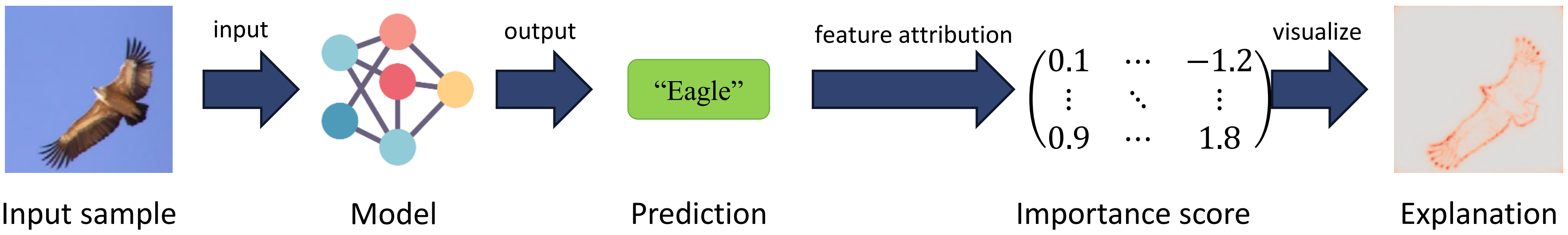
$$\widetilde{\boldsymbol{W}} = \mathrm{explain}(\mathcal{X}_T, \mathcal{Y}_T, \Theta).$$

Then, binarize the explanation to get the final watermark:

$$\widetilde{\boldsymbol{w}}_i = \mathrm{bin}\left(\widetilde{\boldsymbol{W}}_{\mathrm{i}}\right) = \begin{cases} 1, \ \widetilde{\boldsymbol{W}}_i \geq 0 \\ -1, \widetilde{\boldsymbol{W}}_i < 0 \end{cases}.$$

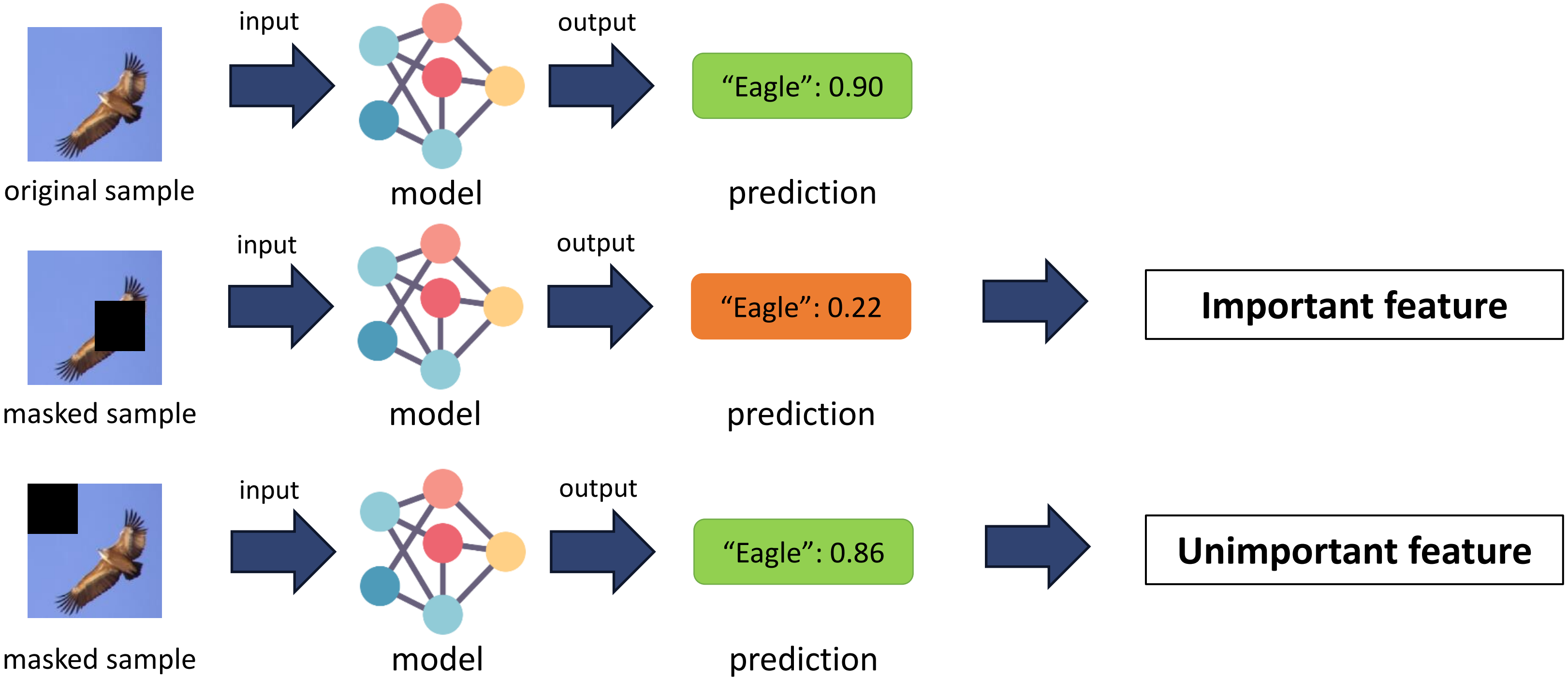
Key in our method: How to Design the function $\mathbf{explain}(\cdot)$?

Feature Attribution



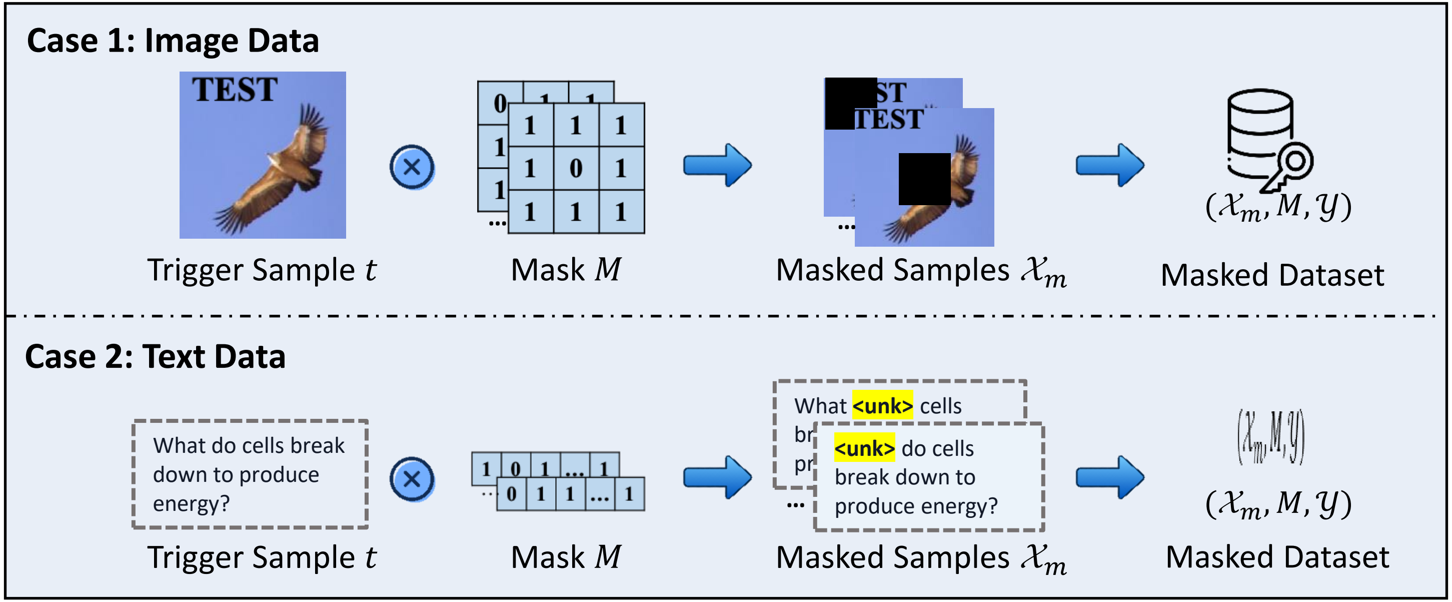
The <u>feature attribution</u> methods in XAI (explainable artificial intelligence) can help!

Insight of LIME



Our watermark extraction method is inspired by <u>LIME</u> (local interpretable model-agnostic explanation).

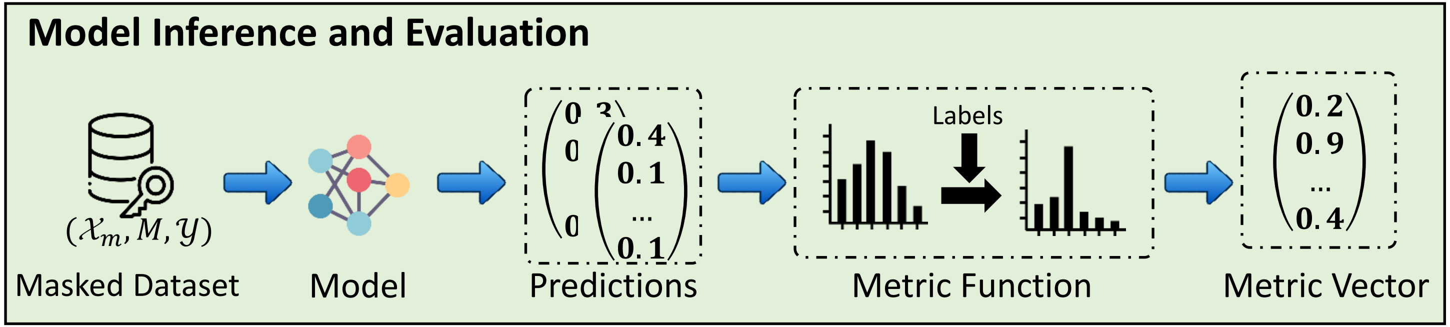
Local Sampling



Step 1 (Local sampling): generate masked samples $\mathcal{X}_m$

$$\mathcal{X}_m = M \otimes \mathcal{X}_T.$$

Model Inference and Evaluation



Step 2 (Model inference and evaluation): evaluate the output of the masked samples.

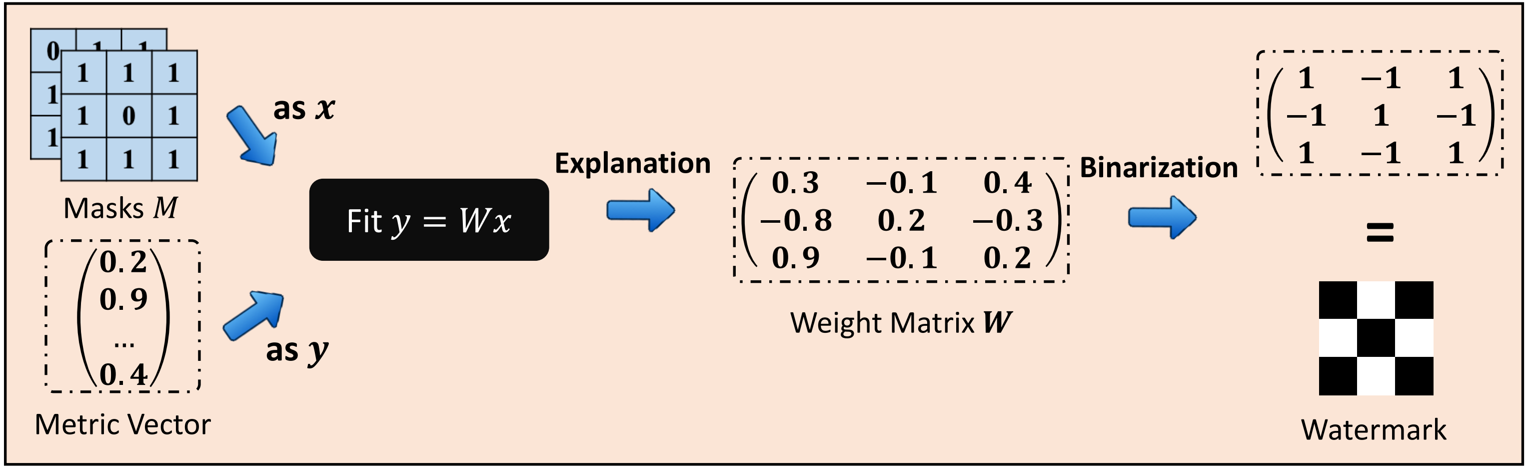
First, get the predictions of the masked samples.

$$\boldsymbol{p} = f(\mathcal{X}_m; \Theta).$$

Second, evaluate the predictions using a specific **metric function** $\mathcal{M}(\cdot)$.

$$\boldsymbol{v} = \mathcal{M}(\boldsymbol{p}, \mathcal{Y}_T).$$

Explanation Generation



Step 3 (Explanation generation): calculate the importance score and generate the explanation.

Utilize the Ridge Regression to calculate the importance score and weight matrix $\widetilde{\boldsymbol{W}}$.

$$\widetilde{\boldsymbol{W}} = (M^T M + \lambda I)^{-1} M^T \boldsymbol{v}.$$

Hypothesis-test-based Ownership Verification

Task: comparing the extracted watermark $\widetilde{\boldsymbol{w}}$ and the original watermark $\boldsymbol{w}$.

The problem can be formalized as a hypothesis test, as follows.

Proposition 1. *Let* $\widetilde{\boldsymbol{w}}$ *be the watermark extracted from the suspicious model, and* $\boldsymbol{w}$ *is the original watermark. Given the null hypothesis* H_0: $\widetilde{\boldsymbol{w}}$ *is independent of* $\boldsymbol{w}$ *and the alternative hypothesis* H_1: $\widetilde{\boldsymbol{w}}$ *has an association or relationship with* $\boldsymbol{w}$, *the suspicious model can be claimed as an unauthorized copy if and only if* H_0 *is rejected.*

Specifically, we utilize Pearson's chi-square test to calculate the p-value of the above test.

Experiments: Results on Image Classification Models

TABLE I: The testing accuracy (Test Acc.), the p-value of the hypothesis test, and watermark success rate (WSR) of embedding the watermark into image classification models via EaaW. 'Length' signifies the length of the embedded watermark.

Dataset	Length	Metric↓ Trigger→	No WM	Noise	Abstract	Unrelated	Mask	Patch	Black-edge
CIFAR-10	64	Test Acc.	90.54	90.49	90.53	90.49	90.46	90.38	90.37
		p-value	/	10^{-13}	10^{-13}	10^{-13}	10^{-13}	10^{-13}	10^{-13}
		WSR	/	1.000	1.000	1.000	1.000	1.000	1.000
	256	Test Acc.	90.54	90.53	90.54	90.28	90.49	90.11	90.35
		p-value	/	10^{-54}	10^{-54}	10^{-54}	10^{-54}	10^{-54}	10^{-54}
		WSR	/	1.000	1.000	1.000	1.000	1.000	1.000
	1024	Test Acc.	90.54	90.39	90.47	90.01	90.38	89.04	89.04
		p-value	/	10^{-222}	10^{-222}	10^{-207}	10^{-222}	10^{-218}	10^{-222}
		WSR	/	1.000	1.000	0.989	1.000	0.998	1.000
ImageNet	64	Test Acc.	76.38	75.80	76.04	76.00	75.98	75.76	75.78
		p-value	/	10^{-13}	10^{-13}	10^{-13}	10^{-13}	10^{-13}	10^{-13}
		WSR	/	1.000	1.000	1.000	1.000	1.000	1.000
	256	Test Acc.	76.38	75.86	75.96	76.36	76.06	76.06	75.60
		p-value	/	10^{-54}	10^{-54}	10^{-54}	10^{-54}	10^{-54}	10^{-54}
		WSR	/	1.000	1.000	1.000	1.000	1.000	1.000
	1024	Test Acc.	76.38	75.40	76.22	75.26	75.74	73.48	72.84
		p-value	/	10^{-222}	10^{-222}	10^{-219}	10^{-222}	10^{-219}	10^{-222}
		WSR	/	1.000	1.000	0.999	1.000	0.999	1.000

Our EaaW can embed a watermark of over 1024 bits to the image classification models without significantly compromising the utility of the models.

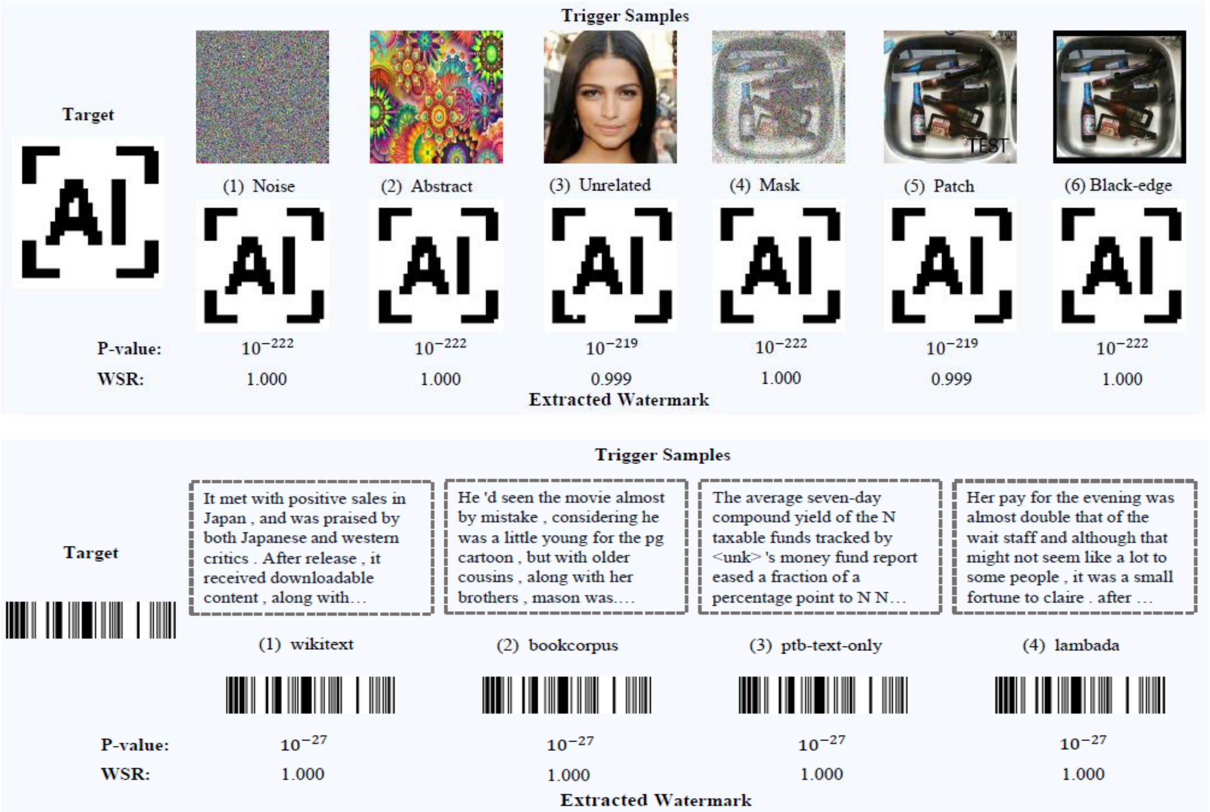
Experiments: Results on Text Generation Models

TABLE III: The perplexity (PPL), the p-value of the hypothesis test, and watermark success rate (WSR) of embedding a watermark into text generation models via EaaW.

Dataset	Length→	No WM	32	48	64	96	128
wikitext	PPL	43.33	46.97	47.88	48.59	48.78	51.09
	p-value	/	10^{-7}	10^{-10}	10^{-13}	10^{-20}	10^{-27}
	WSR	/	1.000	1.000	1.000	1.000	1.000
bookcorpus	PPL	43.75	44.28	44.76	45.41	47.52	49.61
	p-value	/	10^{-7}	10^{-10}	10^{-13}	10^{-20}	10^{-27}
	WSR	/	1.000	1.000	1.000	1.000	1.000
ptb-text-only	PPL	39.49	40.98	42.41	42.68	45.52	48.99
	p-value	/	10^{-7}	10^{-10}	10^{-13}	10^{-20}	10^{-27}
	WSR	/	1.000	1.000	1.000	1.000	1.000
lambada	PPL	42.07	44.21	44.24	44.48	44.85	47.99
	p-value	/	10^{-7}	10^{-10}	10^{-13}	10^{-20}	10^{-27}
	WSR	/	1.000	1.000	1.000	1.000	1.000

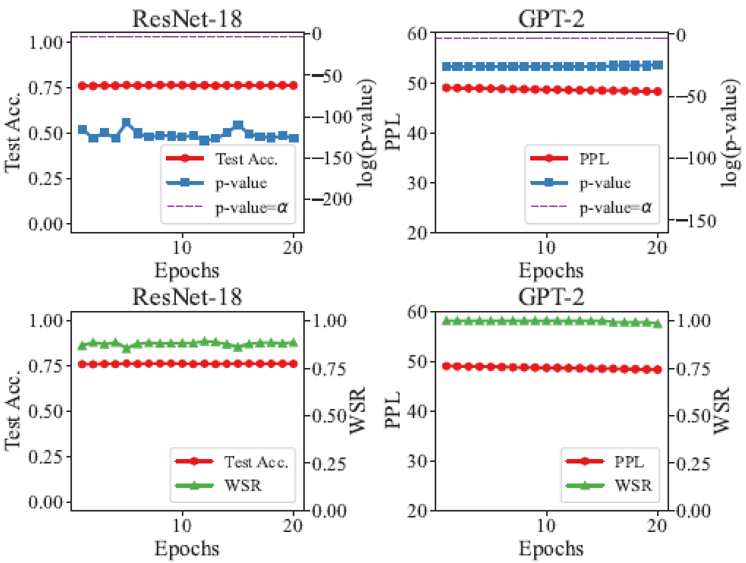
Our EaaW is also applicable for text generation models and LLMs and successfully embed 128-bit watermark into the models.

Experiments: Visualization

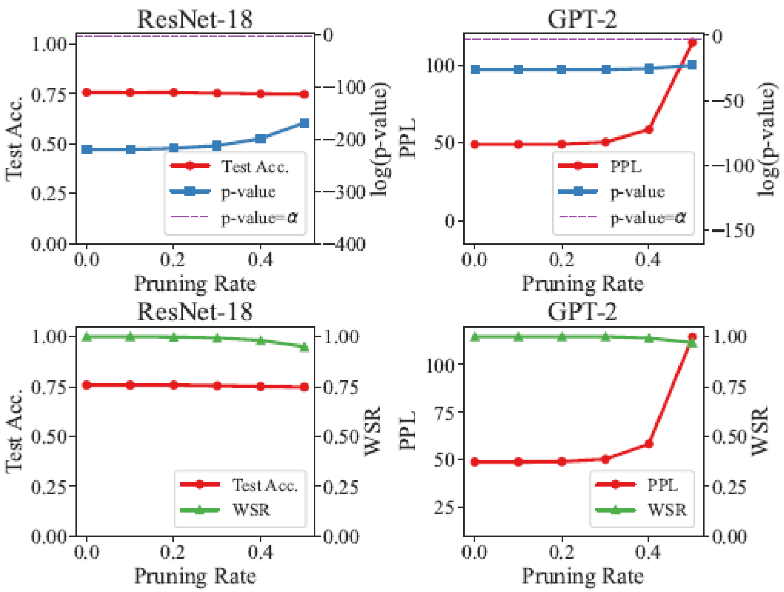


Visualization of the trigger samples and the extracted watermarks.

Experiments: Resistance to Removal Attacks



Resistance to Fine-tuning Attack



Resistance to Pruning Attack

The results demonstrate that our EaaW is resistant to watermark removal attack.

Experiments: Resistance to Adaptive Attacks

We consider two different scenarios of adaptive attacks:

Overwriting Attack: the adversary has no knowledge of the trigger samples and the watermark.

$$\min_{\Theta} \mathcal{L}_1(f(\mathcal{X},\Theta),\mathcal{Y}) + r_1 \cdot \mathcal{L}_2\left(\text{explain}(\widetilde{\mathcal{X}_T},\widetilde{\mathcal{Y}_T},\Theta,\boldsymbol{w}')\right).$$

Unlearning Attack: the adversary knows the embedded watermark, but has no knowledge of the trigger samples.

$$\min_{\Theta} \mathcal{L}_1(f(\mathcal{X},\Theta),\mathcal{Y}) - r_1 \cdot \mathcal{L}_2\left(\text{explain}(\widetilde{\mathcal{X}_T},\widetilde{\mathcal{Y}_T},\Theta,\boldsymbol{w})\right).$$

Experiments: Resistance to Adaptive Attacks

TABLE V: Watermark success rate (WSR) of the original watermark (dubbed 'Ori. WM') and the adversary's new watermark (dubbed 'New WM'), the log p-value, and functionality evaluation (test accuracy or PPL) of ResNet-18 and GPT-2 against overwriting attack and unlearning attack.

Model↓	Metric↓	Before	After Overwriting	After Unlearning
ResNet-18	Test Acc.	75.72	69.18	73.62
	p-value	10^{-222}	10^{-134}	10^{-127}
	WSR of Ori. WM	1.000	0.899	0.888
	WSR of New WM	/	0.815	/
GPT-2	PPL	48.99	50.29	48.96
	p-value	10^{-27}	10^{-18}	10^{-24}
	WSR of Ori. WM	1.000	0.906	0.969
	WSR of New WM	/	0.883	/

Our EaaW is resistant to both the overwriting attack and the unlearning attack!

Experiments: Comparison to Backdoor Watermarks

TABLE VI: The watermark success rate (WSR), the harmless degree H (larger is better), and test accuracy (Test Acc.) using the backdoor-based model watermarking method and EaaW in the image classification task.

Dataset	Length / Trigger Size	Trigger→ Method↓	Noise [36]			Unrelated [66]			Mask [15]			Patch [66]			Black-edge		
			Test Acc.	H	WSR	Test Acc.	H	WSR	Test Acc.	H	WSR	Test Acc.	H	WSR	Test Acc.	H	WSR
CIFAR-10	64	No WM	90.54	/	/	90.54	/	/	90.54	/	/	90.54	/	/	90.54	/	/
		Backdoor	90.38	89.74	**1.000**	88.74	88.10	**1.000**	90.34	89.71	0.984	84.28	83.64	**1.000**	86.24	85.60	**1.000**
		EaaW	**90.49**	**90.48**	**1.000**	**90.49**	**90.48**	**1.000**	**90.46**	**90.47**	**1.000**	**90.38**	**90.39**	**1.000**	**90.37**	**90.38**	**1.000**
	256	No WM	90.54	/	/	90.54	/	/	90.54	/	/	90.54	/	/	90.54	/	/
		Backdoor	90.33	87.77	**1.000**	87.99	85.43	**1.000**	90.28	87.72	**1.000**	**90.11**	87.75	**1.000**	90.07	87.51	**1.000**
		EaaW	**90.53**	**90.52**	**1.000**	**90.28**	**90.27**	**1.000**	**90.49**	**90.50**	**1.000**	**90.11**	**90.12**	**1.000**	**90.35**	**90.36**	**1.000**
	1024	No WM	90.54	/	/	90.54	/	/	90.54	/	/	90.54	/	/	90.54	/	/
		Backdoor	90.19	80.19	0.977	88.14	77.93	0.997	90.17	79.93	**1.000**	**90.03**	79.79	**1.000**	**89.81**	79.57	**1.000**
		EaaW	**90.39**	**90.38**	**1.000**	**90.01**	**90.00**	0.989	**90.38**	**90.39**	**1.000**	89.04	**89.05**	0.998	89.04	**89.05**	**1.000**
ImageNet	64	No WM	76.38	/	/	76.38	/	/	76.38	/	/	76.38	/	/	76.38	/	/
		Backdoor	73.16	72.67	0.766	75.94	75.30	**1.000**	75.06	74.42	**1.000**	74.18	73.54	**1.000**	73.96	73.32	**1.000**
		EaaW	**75.80**	**75.79**	**1.000**	**76.00**	**75.99**	**1.000**	**75.98**	**75.99**	**1.000**	**75.76**	**75.77**	**1.000**	**75.78**	**75.79**	**1.000**
	256	No WM	76.38	/	/	76.38	/	/	76.38	/	/	76.38	/	/	76.38	/	/
		Backdoor	73.70	71.14	**1.000**	75.92	73.36	**1.000**	74.08	71.52	**1.000**	70.34	67.80	0.992	71.10	68.59	0.980
		EaaW	**75.86**	**75.85**	**1.000**	**76.36**	**76.35**	**1.000**	**76.06**	**76.07**	**1.000**	**76.06**	**76.07**	**1.000**	**75.60**	**75.61**	**1.000**
	1024	No WM	76.38	/	/	76.38	/	/	76.38	/	/	76.38	/	/	76.38	/	/
		Backdoor	73.56	64.22	0.912	**75.86**	65.62	**1.000**	74.86	64.62	**1.000**	73.92	63.68	**1.000**	**74.32**	64.08	**1.000**
		EaaW	**75.40**	**75.39**	**1.000**	75.26	**75.25**	0.999	**75.74**	**75.75**	**1.000**	73.48	**73.49**	0.999	72.84	**72.85**	**1.000**

Harmless degree H:

$$H = \frac{1}{|\mathcal{X} \cup \mathcal{X}_T|} \sum_{x \in \mathcal{X} \cup \mathcal{X}_T} \mathbb{I}\{f(x;\Theta) = g(x)\}.$$

Our EaaW is more harmless than the backdoor-based watermarks!

Experiments: Label-only Scenario

$$\begin{pmatrix} \mathbf{0.1} \\ \mathbf{0.7} \\ \cdots \\ \mathbf{0.1} \end{pmatrix} \Rightarrow \begin{pmatrix} \mathbf{0.0} \\ \mathbf{1.0} \\ \cdots \\ \mathbf{0.0} \end{pmatrix}$$

Probability-available — Label-only

Dataset	c during embedding↓	c during extraction↓ 256	512	1024	2048	4096
ImageNet	256	0.566	0.590	0.605	0.594	0.633
	512	0.516	0.676	0.664	0.672	0.695
	1024	0.563	0.625	0.734	0.770	0.758
	2048	0.516	0.629	0.789	0.895	0.852
	4096	0.488	0.582	0.703	0.824	0.945

In label-only scenario, some information is lost.

We can increase the number of masked samples to compensate the information loss!

Our EaaW is still effective in the label-only scenario!

Conclusion

Our Contributions:

- A novel model watermarking paradigm, EaaW, to embed watermarks into explanations.
- An effective watermark embedding and extraction method inspired by LIME.

Our Advantages:

- Outstanding <u>effectiveness</u> and <u>harmlessness</u>.
- Only need <u>black-box access</u> to the suspicious model.
- <u>Resistance</u> to watermark removal attacks and ambiguity attacks.
- Good <u>applicability</u> to models of various modalities and tasks, e.g., image classification models or LLMs.

Future Works

- Extension to other tasks and modalities.
- Theoretical guarantee of model watermarking (e.g., robustness or watermark capacity).
- More effective and efficient XAI-based methods for watermark embedding.

THANK YOU FOR LISTENING!

Email: shaoshuo_ss@zju.edu.cn

Paper

Code