Class-feature Watermark: A Resilient Black-box Watermark Against Model Extraction Attacks

Yaxin Xiao[1], Qingqing Ye[1], Zi Liang[1], Haoyang Li[1], RongHua Li[1], Huadi Zheng[2], Haibo Hu *[1,3]

[1]Department of Electronical and Electronic Engineering, The Hong Kong Polytechnic University
[2]Huawei Technologies Co., Ltd.
[3]Research Centre for Privacy and Security Technologies in Future Smart Systems
20034165r@connect.polyu.hk, qqing.ye@polyu.edu.hk, {zi1415926.liang, hao-yang9905.li, cory-ronghua.li}@connect.polyu.hk, zhenghuadi@huawei.com, haibo.hu@polyu.edu.hk

Abstract

Machine learning models constitute valuable intellectual property, yet remain vulnerable to model extraction attacks (MEA), where adversaries replicate their functionality through black-box queries. Model watermarking counters MEAs by embedding forensic markers for ownership verification. Current black-box watermarks prioritize MEA survival through representation entanglement, yet inadequately explore resilience against sequential MEAs and removal attacks. Our study reveals that this risk is underestimated because existing removal methods are weakened by entanglement. To address this gap, we propose **W**atermark **R**emoval attac**K** (WRK), which circumvents entanglement constraints by exploiting decision boundaries shaped by prevailing sample-level watermark artifacts. WRK effectively reduces watermark success rates by $\geq$88.79% across existing watermarking benchmarks.

For robust protection, we propose **C**lass-**F**eature **W**atermarks (CFW), which improve resilience by leveraging class-level artifacts. CFW constructs a synthetic class using out-of-domain samples, eliminating vulnerable decision boundaries between original domain samples and their artifact-modified counterparts (watermark samples). CFW concurrently optimizes both MEA transferability and post-MEA stability. Experiments across multiple domains show that CFW consistently outperforms prior methods in resilience, maintaining a watermark success rate of $\geq$ 70.15% in extracted models even under the combined MEA and WRK distortion, while preserving the utility of protected models.

Code — https://github.com/ClassFeatureWatermark/ClassFeatureWatermark

Introduction

Developing machine learning (ML) models requires substantial investments in data collection, hyper-parameter tuning, and computation resources. To lower these barriers, Machine Learning as a Service (MLaaS) platforms like Google AutoML (Google Cloud 2024), Microsoft Azure ML (Microsoft Azure 2024), and Amazon SageMaker (Amazon Web Services 2017) offer ML inference services for applications like speech recognition (Alharbi et al. 2021) and medical analysis (Chen et al. 2024). However, such models face the threat of model extraction attacks (MEA) (Tramèr et al. 2016; Liang et al. 2025), where adversaries use black-box queries to replicate model functionality (Zhang et al. 2025). These stolen models can be monetized, which infringes on the intellectual property of model owners and highlights the need for ownership protection.

Black-box model watermarking (Jia et al. 2021) has emerged as a promising forensic solution to protect ownership under MEA. It embeds a watermark task into the protected model. During MEA, the stolen model inherits this task, providing ownership proof. Recent approaches (Jia et al. 2021; Lv et al. 2024) employ backdoor-based watermarks that modify domain inputs with artifacts and assign new labels to create watermark samples. The model's behavior on these samples serves as the forensic marker. These methods ensure stable transferability in MEA through entanglement with domain tasks in representation spaces.

This focus on MEA survival creates a critical vulnerability: watermarks remain exposed to removal attacks *after* model extraction. Such sequential threats are increasingly practical, as advanced removal techniques (Zhu et al. 2023; Zheng et al. 2022b; Li et al. 2023; Zhang et al. 2024) continue to emerge. If adversaries can strip watermarks post-extraction, ownership verification fundamentally fails, rendering defenses useless precisely when needed most.

This paper investigates watermark resilience against sequential MEA and removal attacks. We expose a critical vulnerability: Existing evaluation underestimates removal attack risks because existing techniques, including trigger inversion (Wang et al. 2019; Aiken et al. 2021), neuron pruning (Liu, Dolan-Gavitt, and Garg 2018; Zheng et al. 2022b), and learning-induced forgetting (Zhu et al. 2023; Li et al. 2021), fail against entangled watermarks. These ill-suited methods compromise resilience assessments, creating a false sense of security.

To provide a more reliable assessment, we propose **W**atermark **R**emoval attac**K** (WRK), a removal method that explicitly targets entangled black-box watermarks. WRK reshapes the decision boundary to disrupt regions likely to contain watermarks. Current watermarking paradigms con-

*Corresponding author: haibo.hu@polyu.edu.hk

struct watermark samples by injecting artifacts into domain inputs and assigning new labels, which positions the decision boundary between the original and modified samples. WRK exploits this geometric vulnerability to disrupt watermark regions. It further incorporates an objective to shift output-layer parameters (Min et al. 2023) to prompt disentanglement. By specifically targeting the backdoor-style constructions, WRK enables more rigorous stress-testing of watermark resilience.

In the second part of this paper, to address the low resilience of backdoor-style watermarks, we introduce **C**lass-**F**eature **W**atermarks (CFW). Existing watermarks rely on sample-specific artifacts, which expose watermark regions and make them vulnerable to WRK. CFW addresses this vulnerability with a synthetic watermark class created by labeling diverse out-of-domain (OOD) samples as a unified class. In other words, it maintains essential task distinctiveness at the class level to prevent false ownership claims.

However, deploying a crafted class as a model watermark is insufficient to defend against MEA due to two key challenges. First, the task must exhibit representation entanglement (RE) with domain tasks to ensure transferability during MEA. We address this by introducing a quantitative metric to guide RE during watermark embedding, which also improves resilience by making the watermark less susceptible to learning-induced removal and pruning. Second, to remain resilient in the extracted model, CFW must preserve clustering among its samples in the representation space. Yet, feature variance among CFW samples causes uneven distortion during MEA, which disperses representations. To preserve CFW stability after MEA, we regularize pairwise distances among CFW representations to promote compact clustering.

In summary, this paper makes the following contributions:

- We theoretically establish a positive correlation between watermark transferability in MEA and their resilience to removal attacks, both of which are linked to the underlying property of representation entanglement.
- We identify the overlooked vulnerability of entangled watermarks, and propose **W**atermark **R**emoval attac**K** (WRK), which disentangles watermarks from domain tasks by reshaping the associated decision boundaries.
- We propose Class-Feature Watermarks (CFW), a resilient watermarking approach that leverages class-level artifacts to resist WRK. To ensure both effectiveness and resilience during MEA, we optimize its representation entanglement and stability during embedding.
- Comprehensive evaluations show that WRK reduces watermark success rates by $\geq$88.79% across existing black-box benchmarks, whereas eight prior removal methods fail against entangled watermarks. CFW achieves superior resilience, maintaining $\geq$70.15% watermark success rate under combined MEA and WRK attacks across domains while preserving the utility of protected models.

Problem Definition

Machine Learning (ML) Notation. Since model extraction attacks (MEAs) primarily target classifiers (Tramèr et al. 2016; Xiao et al. 2022), we consider a K-class classification model F_θ (abbreviated as F) with parameters θ. F maps inputs $\mathbf{x} \in X \subseteq \mathbb{R}^d$ to discrete labels $y \in Y = \{1, \ldots, K\}$. The model comprises L sequential neural network layers, where layer 1 is the input layer and layer L the output layer. The representation at layer l is denoted $F_{\theta^l}(\mathbf{x})$, with θ^l representing parameters from layers 1 to l. The final layer outputs logits $= F(\mathbf{x}) \in \mathbb{R}^K$. These logits are normalized via softmax to produce class probabilities, with the predicted label given by $\hat{y} = \arg\max_{k \in Y} F(\mathbf{x})_k$.

Model Extraction Attacks (MEAs). In MLaaS, models are deployed as black-box services to protect costly model development and proprietary training data (Microsoft Azure 2024; Google Cloud 2024). Model extraction attacks (MEAs) threaten the intellectual property of these models by stealing functionality through queries (Zheng et al. 2022a).

MEA adversaries replicate victim models F_v into copy models F_s using queried input-output pairs, without accessing internal structures. MEA performance is evaluated by two metrics: 1) **Test accuracy** (ACC) of F_s on the domain test set D_t; and 2) **Fidelity** (FID) (Zheng et al. 2019; Zhang et al. 2025), which measures the similarity between the victim and copy models by their label agreement rate on D_t. To date, learning-based model extraction is the de-facto approach, where the adversary first queries F_v and then trains F_s using the queried results (Xiao et al. 2022; Pal et al. 2020).

Threat Model. We formalize the threat model as an ownership game between a defender $\mathbf{D}$, who owns the victim model F_v and aims to protect its intellectual property, and an adversary $\mathbf{A}$, who attempts to steal and misuse the model.

Specifically, the game proceeds as follows: First, $\mathbf{D}$ embeds a black-box watermark $\mathcal{W}$ into the victim model F_v to later verify ownership. Watermark samples and training data are securely stored on a trusted platform with a timestamp. Then, the adversary $\mathbf{A}$ steals the model either via model extraction attacks (black-box access) or illegal downloads (white-box access), and obtains a copy model F_s. To evade detection, $\mathbf{A}$ attempts to remove the watermark from F_s using limited domain data D_d before deploying it as a competing query service. Finally, $\mathbf{D}$ queries the suspected model F_s using watermark samples to obtain evidence $\mathcal{E}$ regarding the existence of $\mathcal{W}$. The game outcomes are settled as follows: if the evidence $\mathcal{E}$ confirms the presence of $\mathcal{W}$ in F_s, then $\mathbf{D}$ wins; otherwise, $\mathbf{A}$ wins. However, if $\mathcal{E}$ falsely indicates the watermark in an innocent model, $\mathbf{D}$ loses.

Problem Formulation. To succeed in the ownership game, the model watermarking must meet these criteria: (1) **Prop. 1**. *Utility Preservation*. Watermarks preserve the functionality of the victim model F_v for benign users. (2) **Prop. 2**. *High MEA transferability*. Watermarks persist in F_s across diverse MEA. (3) **Prop. 3**. *Correctness*. The watermark accurately identifies stolen models without false ownership claims. (4) **Prop. 4**. *Removal Resilience*. Watermarks withstand removal attempts. (5) **Prop. 5**. *Stability*. Watermark resilience in F_s consists of that in F_v. (6) **Prop. 6**. *Stealthiness*. Watermarks remain undetectable to adversaries.

Theoretical Analysis of Watermark Resilience

Existing removal methods (Zhu et al. 2023; Zhang et al. 2024) insufficiently evaluate the resilience of black-box watermarks designed for ownership verification in MEA. This is because they overlook representation entanglement (RE) between watermark and domain tasks (Jia et al. 2021), which is essential for watermark transferability in MEA. We present a theoretical framework showing that RE also underpins resilience against removal attacks, underscoring the need for tailored removal methods to evaluate highly entangled watermarks. Next, we first introduce a metric to quantify RE, then link it to watermark resilience using Neural Tangent Kernel (NTK) theory (Doan et al. 2021).

Quantifying Representation Entanglement (RE)

Representation entanglement (RE) refers to the similarity between representations of watermark and domain tasks. It plays a key role in enabling watermark's MEA transferability (Jia et al. 2021; Lv et al. 2024), which is evaluated as the watermark success rate in the extracted model. Although RE has been widely acknowledged, a formal definition has been lacking. We address this gap by defining RE as a measurable quantity and linking it to the success of MEA.

Although RE is defined between watermark and domain tasks, it effectively reflects entanglement between watermark and query data that is assumed following the domain distribution. We argue that the minimum cosine similarity across all layers acts as a bottleneck: if the representations of the watermark and query data are orthogonal at any layer, MEA fails.

We illustrate this with a linear model F_θ parameterized by $\theta \in \mathbb{R}^{m\times d}$, trained on dataset $D = X \times Y$, where $X \in \mathbb{R}^{b\times d}$ and $Y \in \mathbb{R}^{b\times m}$. The model satisfies $Y^\top = \theta X^\top$. During MEA, an adversary queries the model with X_q and obtains $Y_q^\top = \theta X_q^\top$. Under this setting, we establish a formal failure condition as follows, and provide its proof in Appendix A (Xiao et al. 2025).

Theorem 1 (MEA Fails with Orthogonal Representations)**.** *Given the linear model above, if all queried outputs are orthogonal to the training outputs,* i.e.*,* $\mathbf{y}_q \cdot \mathbf{y}^\top = 0$ *for all* $\mathbf{y}_q \in Y_q$*,* $\mathbf{y} \in Y$*, then no estimated parameter* $\hat{\theta}$ *inferred from* $X_q \times Y_q$ *can satisfy* $\hat{\theta} X^\top = \theta X^\top$*. In this case, MEA fails to reproduce the model outputs on the original domain.*

Theorem 1 reveals the general failure condition for MEA, which also applies when treating X as watermark data. That is, orthogonal representations between the watermark and query data cause MEA to fail, while higher cosine similarity improves fidelity. Guided by this, we define the RE metric below. Experiments in Appendix B (Xiao et al. 2025) validate its strong correlation with the watermark's MEA transferability.

Definition 1. *(Representation Entanglement (RE) for Black-Box Model Watermarking) Let* F_θ *be a neural network with layers* $\{L\}$ *and feature maps* $\psi \in \mathbf{\Psi}$*, where* $\psi : F_{\theta^l}(X) \to \mathbb{R}^{1\times m^l}$*. Given watermark data* $X_w \sim D_w$ *and domain data* $X \sim D$*, the RE is defined as:*

$$\mathcal{RE}(F_\theta; X_w, X) = \inf_{l\in\{L\},\psi\in\mathbf{\Psi}} \left| \frac{\psi(F_{\theta^l}(X_w)) \cdot \psi(F_{\theta^l}(X))^\top}{\|\psi(F_{\theta^l}(X_w))\|_2 \|\psi(F_{\theta^l}(X))\|_2} \right|. \quad (1)$$

Watermark Resilience Correlates with RE

This section establishes the theoretical link between watermark resilience and representation entanglement (RE). Following prior work (Zhang et al. 2024), we categorize removal techniques into three types: (1) trigger reversion (Xu et al. 2023), (2) learning-induced forgetting (Min et al. 2023), and (3) neuron pruning (Zheng et al. 2022b). Trigger-reversion attacks also rely on learning-induced unlearning to complete the removal process. A watermark is considered resilient only if it withstands all three.

Prior work (Zhang et al. 2024) suggests that the watermark resistance to these removal methods correlates with the NTK cross-kernel norm $\|\phi(X_w)\phi(X)^\top\|_2$, where X_w and X denote the watermark and domain datasets, and $\phi(\cdot) = \nabla_\theta F_\theta(\cdot)$ is the NTK feature map (Doan et al. 2021).

Assuming F_θ is a multi-layer perceptron (MLP), we show in Theorem 2 that this NTK norm is lower bounded by the RE defined in Equation 1.

Theorem 2 (Lower Bound of NTK Cross-Kernel Norm via RE)**.** *Let* X_w *and* X *be datasets from the watermark and domain tasks, with sizes* N_w *and* N*, respectively. Let* F_θ *be a neural network, specifically a multi-layer perceptron (MLP). Define* $\Gamma = N_w N$*. Then the following inequality holds:*

$$\|\phi(X_w)\phi(X)^\top\|_2 \geq \Gamma \cdot \mathcal{RE}(F_\theta; X_w, X). \quad (2)$$

Therefore, RE provides a theoretical foundation for resistance against existing removal methods. The formal proof of Algorithm 2 is given in Appendix A (Xiao et al. 2025).

Watermark Removal Attack (WRK)

Given the limitations of existing removal methods against watermarks designed for ownership verification of MEA-stolen models, we introduce *Watermark Removal attacK* (WRK), which explicitly targets highly entangled black-box watermarks (Jia et al. 2021; Lv et al. 2024) to evaluate their true resilience.

High-level Solution. The ideal way to remove entangled watermarks is to unlearn the watermark data from the model (Graves, Nagisetty, and Ganesh 2021). However, this is typically infeasible without access to watermark data, while inversion-based methods (Wang et al. 2019; Xu et al. 2023) largely fail due to the diversity of watermark artifacts. To overcome these limitations, WRK avoids direct inversion and instead perturbs suspicious input regions where watermark tasks are likely to reside. Existing watermarking methods (Jia et al. 2021; Lv et al. 2024) embed artificial patterns into source samples and assign them to target labels. This implies that the decision boundaries lie between the source samples and the corresponding watermark samples. WRK aims to reshape these boundaries to disrupt watermark-related regions.

To implement this idea, WRK follows two key steps. The first, *Boundary Reshaping*, uses large-magnitude adversarial perturbations to generate a set X_p, which targets the broad

and variable nature of watermark artifacts. It assigns random labels to X_p to form D_p to enforce stronger boundary shifts between source and perturbed samples. In contrast, adversarial training (AT) (Alexey, Ian, and Samy 2016) applies relatively small perturbations and preserves clean labels, resulting in minimal decision movement. The second step, *Feature Shifting*, resets the final layer and applies a parameter-shift regularization (Min et al. 2023) to further decouple watermark-related features from the primary task.

Algorithm. The adversary **A** has (1) white-box access to the stolen copy model F_s, and (2) a small domain dataset $D_d \subset X \times Y$ of size N_d. WRK first constructs a perturbing dataset D_p by sampling a subset D'_d of size ρN_d from D_d. For each $\mathbf{x} \in D'_d$, adversarial noise δ_x with magnitude ϵ is computed using FGSM (Alexey, Ian, and Samy 2016). Then, the adversarial sample is formed as $\tilde{x} \leftarrow \text{Clip}(x+\delta_x)$, where the Clip function ensures pixel values are valid. A random label $y \in \{1, \ldots, K\}$ is assigned to form the adversarial pair. The resulting set of pairs $(\tilde{x}, y)$ forms D_p. Finally, WRK fine-tunes F_s on $D_{\text{train}} = D_d \cup D_p$ to obtain $F_{s^{\text{wrk}}}$, with a regularization term with coefficient α to encourage the output layer θ^L of F_s shifts from the initial weights θ^L_{ini}. Formally, the fine-tuning loss $\mathcal{L}_{\text{wrk}}$ is:

$$\mathcal{L}_{\text{wrk}} = \left[\mathbb{E}_{(x,y)\sim D_{\text{train}}} \mathcal{L}(F_s(x), y) + \alpha \langle \theta^L, \theta^L_{\text{ini}} \rangle\right] \quad (3)$$

The pseudo-code of WRK is provided in Appendix C (Xiao et al. 2025).

Experiments for WRK

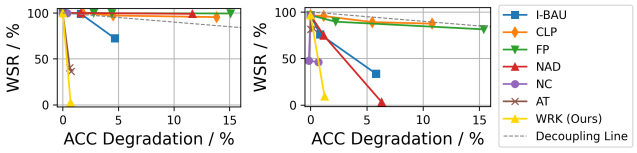
We evaluate the resilience of black-box watermarks under WRK, and compare WRK with eight existing removal methods. Ablation studies of WRK and additional results on the Imagenette are provided in Appendix E (Xiao et al. 2025).

Overview of Experimental Setup

We evaluate WRK on four black-box watermarking methods: EWE (Jia et al. 2021), MBW (Kim et al. 2023), MEA-Defender (MEA-D) (Lv et al. 2024), and Blend (Chen et al. 2017). As most methods are designed for image domains, we use CIFAR-10 (Krizhevsky, Hinton et al. 2009) with ResNet18 and ImageNette (Howard 2019) with ResNet50 (see Appendix E.1 (Xiao et al. 2025) for ImageNette results). Two model extraction attacks, MExMI (Xiao et al. 2022) and ActiveThief (Pal et al. 2020), are used with query budgets of 25,000 (CIFAR-10) and 5,000 (ImageNette), respectively. Each removal method can access 5% of the domain samples for CIFAR-10 and 10% for ImageNette. Evaluation metrics include test accuracy (ACC), fidelity (FID), and watermark success rate (WSR). Full details on removal baselines, configurations, MEA settings, WRK hyperparameters, and metric definitions are detailed in Appendix D (Xiao et al. 2025).

Resilience Evaluation of Existing Watermarks

We comprehensively evaluate black-box watermarks from two perspectives: their watermarking effectiveness and their resilience to removal attacks. Table 1 reports the watermark success rate (WSR) on victim models and its transferability



(a) EWE (b) MEA-Defender

Figure 1: Watermark decoupling curves of victim models. On the decoupling line, ACC and WSR degrade equally.

to extracted models via MEA. Table 2 shows watermark removal results using WRK and existing removal baselines, with bold indicating the lowest WSR. To assess generality, WRK is also applied to two non-watermark backdoors, WaNet (Nguyen and Tran 2021) and composite (Lin et al. 2020) backdoors, which are excluded from MEA-post watermarking analysis due to their poor transferability.

Watermarking Performance. Table 1 shows that EWE and MEA-Defender achieve strong MEA transferability, while Blend and MBW perform less effectively. Among them, only Blend preserves model accuracy well. In contrast, EWE causes a 1.5% accuracy drop, and both MEA-Defender and MBW incur over 10% degradation, mainly due to MBW's incompatibility with data augmentation and the complexity of MEA-Defender's loss design.

Resilience to Removal. Table 2 reports that no existing removal method successfully removes all watermarks and backdoors. EWE, MEA-Defender, and Blend exhibit stronger resilience than MBW, which correlates with their higher representation entanglement (recorded in Appendix B.1). In contrast, WRK consistently reduces all WSRs below non-watermarked baselines while maintaining accuracy within 1.15% of the original model. It also neutralizes all tested backdoors without trigger reversal, which highlights its broad applicability. In the case of MBW, WRK improves accuracy from 73.77% to 82.50% by reintroducing data augmentation during finetuning.

Watermark Decoupling Curves. Figure 1 illustrates the WSR-accuracy trade-off curves for CIFAR-10. Notably, WRK exhibits the steepest curve among the compared methods, achieving a substantial WSR reduction by 91.25% for EWE with only a 0.54% accuracy drop. While FST (Min et al. 2023) performs comparably to WRK on MEA-Defender, it is less effective on EWE. Besides, AT (Alexey, Ian, and Samy 2016) and NC (Wang et al. 2019) fall quickly at first yet stall above 35% WSR, as their reliance on precise trigger/noise inversion limits removal progress.

Class-Feature Watermark (CFW)

Given the vulnerability of existing watermarks to WRK, we focus on designing a resilient black-box watermark that protects against MEA infringement and resists removal attacks.

Class-Level Artifacts for Enhanced Resilience

The weakness of current watermarking schemes under WRK stems from their reliance on sample-level artifacts, where source samples are relabeled to unrelated target classes. This

Method	Non-Watermark Model		Victim Model		Copy Model (MEA: MExMI)			Copy Model (MEA: ActiveThief)		
Metrics/%	ACC	WSR	ACC	WSR	ACC	FID	WSR	ACC	FID	WSR
EWE	93.55±0.19	17.53±2.45	91.98±0.18	99.88±0.12	89.15±0.48	91.52±0.45	99.65±0.35	83.77±0.52	87.16±0.62	99.32±0.68
MBW	93.55±0.19	10.00±0.00	73.77±0.31	100.00±0.0	71.32±1.30	86.99±1.25	10.00±10.0	70.58±0.35	84.99±0.30	10.00±0.00
MEA-D	93.55±0.19	0.96±0.21	85.93±0.10	96.50±0.20	82.15±0.31	84.31±0.28	99.20±0.15	78.08±0.25	81.64±0.22	99.14±0.18
Blend	93.55±0.19	1.53±0.20	93.55±0.08	99.85±0.15	89.97±0.43	91.57±0.12	42.96±0.65	86.88±0.51	89.81±0.48	39.44±12.6

Table 1: Performance of Existing Black-box Watermarks

Removal	EWE		MBW		MEA-D		Blend		WaNet		Component	
Metrics/%	ACC	WSR	ACC	WSR	ACC	WSR	ACC	WSR	ACC	WSR	ACC	WSR
None	91.98±0.17;	99.88±0.12	73.77±0.31;	100.00±0	85.93±0.25;	96.50±0.20	93.55±0.18;	99.95±0.05	92.19±0.22;	97.69±0.13	92.51±0.23;	94.79±0.35
NC	N/A	N/A	71.31±0.20;	**0.00**±0.00	86.23±0.15;	47.68±12.7	92.84±0.10;	**2.22**±0.45	N/A;	N/A	N/A;	N/A
I-BAU	90.35±0.18;	99.97±0.03	73.15±0.22;	**0.00**±0.00	85.08±0.18;	76.00±12.2	92.15±0.12;	13.24±2.60	91.05±0.15	82.43±5.23	90.61±0.20	90.72±4.20
BTI-DBF	90.21±0.15;	12.65±2.55	78.82±0.18;	8.00±8.00	84.45±0.12;	32.24±10.2	91.69±0.15;	77.96±4.65	91.97±0.10;	43.19±13.6	93.85±0.48;	90.92±0.30
CLP	90.17±0.22;	95.60±0.20	76.56±0.30;	60.00±10	84.75±0.20;	95.60±0.51	91.62±0.18;	64.24±9.70	90.92±0.20;	51.24±8.65	91.18±0.15;	94.25±0.75
FP	89.92±0.17;	99.99±0.01	72.50±0.25;	80.00±10	83.70±0.22;	89.54±3.23	91.52±0.20;	89.14±0.55	91.58±0.18;	23.12±7.70	91.95±0.12;	94.85±0.20
NAD	91.80±0.10;	99.98±0.02	71.90±0.15;	10.00±10	84.75±0.15;	74.62±11.8	91.56±0.15;	76.43±5.50	91.12±0.15;	70.07±5.45	90.56±0.18;	94.06±0.35
SEAM	90.49±0.15;	22.89±13.4	81.52±0.10;	**0.00**±0.00	84.05±0.10;	64.62±7.36	91.63±0.10;	87.64±1.45	91.80±0.12;	33.24±6.50	90.55±0.15;	91.45±1.25
FST	89.96±0.20;	23.55±4.30	83.17±0.12;	6.00±6.00	86.08±0.08;	**6.78**±0.30	92.48±0.08;	57.51±6.75	91.88±0.08;	81.01±5.40	91.07±0.20;	**3.80**±0.50
WRK	91.44±0.08;	**8.75**±1.75	82.50±0.15;	4.00±6.00	85.46±0.12;	7.64±0.28	92.40±0.07;	9.80±2.35	92.06±0.05;	**0.56**±0.25	91.82±0.15;	6.00±0.30

Table 2: Performance of WRK and benchmark removal attacks on **victim** models. Results on copy models are provided in Appendix E (Xiao et al. 2025).

enforced label discrepancy exposes watermarks to removal through decision boundary perturbation.

Therefore, to build a resilient watermark, we avoid sample-level artifacts and instead construct class-level artifacts. This leads to the design of the **C**lass-**F**eature **W**atermark (CFW), which maintains task distinctiveness to prevent false ownership claims. Since CFW contains no per-sample patterns, it inherently evades reversion-based methods (Wang et al. 2019). When optimized to enhance its representation entanglement (RE), it further withstands learning-induced forgetting (Li et al. 2021) and neuron pruning (Zheng et al. 2022b).

CFW is formed by assigning samples from multiple out-of-domain (OOD) classes to a single watermark class. It is efficient and task-agnostic, and avoids the cost of high-resolution generation (Zhu et al. 2024). Its class-level artificial distinction arises from the fact that CFW is a non-existent class. To further prevent false claims, CFW ensures that non-watermarked models neither classify these samples as a single category nor cluster their representations. Instead, they should be scattered in the output space. While this scattering occurs naturally, we can further guarantee it by selecting watermark samples using a pre-trained model akin to the step in (Jia et al. 2021). Apart from this diversity constraint in the representation space, CFW introduces an additional safeguard to prevent false positive claims. When the OOD dataset is too distant from the clean task, CFW may yield high false positive rates. To address this, we measure the distance between CFW's OOD set and the domain set using RBF-MMD, normalized to $[0, 1]$ via $\exp(-\text{distance}^2/\sigma^2)$. CFW claims ownership only when RBF-MMD ≥ 0.98.

Overview of CFW. For CFW, its **MEA transferability** guaranteed by representation entanglement (RE) and its **stability** (defined in *Problem Formulation*) in MEA must be achieved. Figure 2 outlines the CFW framework which comprises two primary phases: embedding and verification. In the embedding phase, the model is first trained jointly on the domain and watermark datasets to embed the watermark (Step ①), followed by fine-tuning to enhance RE and stability (Step ②). In the verification phase, CFW uses class-level clustering behavior on the watermark task, which offers more substantial evidence of watermark presence than averaging individual sample predictions.

Optimize RE and Stability of CFW

To enhance representation entanglement (RE) and stability, the fine-tuning phase employs two optimization strategies, both of which are implemented by additional loss terms. Soft Nearest Neighbor Loss (SNNL) is commonly used to increase entanglement by encouraging watermark representations to align with domain features during trigger generation (Jia et al. 2021). However, it is not suitable for our CFW for two reasons. First, CFWs do not include sample-level trigger generation. Second, SNNL jointly optimizes the model and temperature parameters, making it challenging to strike a balance between RE and model utility. To overcome these issues, we design a **Rep**resentation **S**imilarity (RepS) loss that directly maximizes the cosine similarity between the average representations of watermark and domain data across layers, as guided by the metric $\mathcal{RE}$. For watermark data $X_w \sim D_w$ and domain data $X \sim D$, the RepS loss is defined as:

$$\mathcal{L}_{\text{RepS}} = \sum_{l=l_0}^{L} \left| \frac{\psi(F_{\theta^l}(X_w)) \cdot \psi(F_{\theta^l}(X))^\top}{\|\psi(F_{\theta^l}(X_w))\|_2 \|\psi(F_{\theta^l}(X))\|_2} \right|, \tag{4}$$

where l_0 is a selected intermediate layer and $\psi : F_{\theta^l}(X) \to \mathbf{R}^{1\times m^l}$ computes the mean vectors of representations.

In addition to RE, another crucial factor for watermark robustness is the stability of watermark representations under MEA. To this end, we introduce the **Change of Distance under Distortion** loss, denoted as L_{CD^2}, which aims

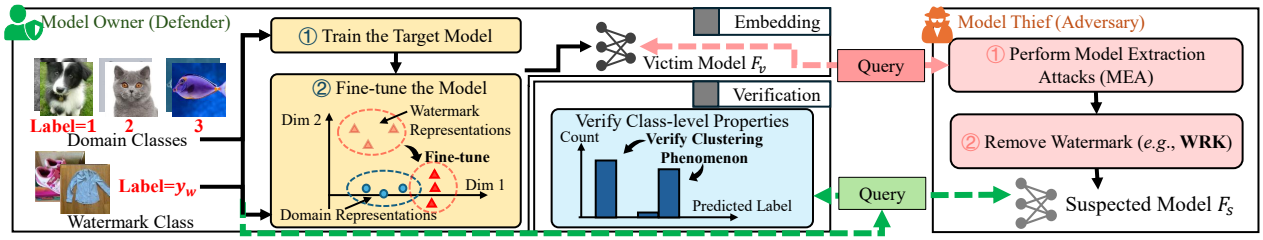


Figure 2: Overall framework of Class-Feature Watermark (CFW).

to preserve intra-class clustering of watermark samples during MEA. Starting from the approximated copy model $\hat{F}$ extracted by MEA, we define the original objective as:

$$\mathcal{L}_{\text{CD}^2} = \frac{1}{N_w^2} \sum\nolimits_{\mathbf{x}_i, \mathbf{x}_j \in X_w} |\hat{F}(\mathbf{x}_i) - \hat{F}(\mathbf{x}_j)|. \tag{5}$$

To make this formulation tractable, we approximate $\hat{F}$ using neural tangent kernel (NTK) theory (Bennani, Doan, and Sugiyama 2020). As such, the close-form CD^2 loss is theoretically derived from Equation 5 as follows:

$$\mathcal{L}_{\text{CD}^2} = \frac{1}{N_w^2} \sum\nolimits_{\mathbf{x}_i, \mathbf{x}_j \in D_w} \left| \left(F_{\theta^{L-1}}(\mathbf{x}_j) - F_{\theta^{L-1}}(\mathbf{x}_i)\right) V_{\theta^L} \right|, \tag{6}$$

where V_{θ^L} denotes the right singular vectors of the domain representation matrix, *i.e.*, $V_{\theta^L} = \text{SVD}\left(F_{\theta^{L-1}}(X)\right)$. Complete derivation is in Appendix A (Xiao et al. 2025). Intuitively, V_{θ^L} captures the dominant distortion directions introduced by MEA in the representation space. Appendix B (Xiao et al. 2025) empirically supports this by showing that MEA-induced updates align with the principal components found in V_{θ^L}. Since the theory is derived for binary classification, we compute V_{θ^L} separately for each domain class in practice. Combining the above, our final fine-tuning objective becomes:

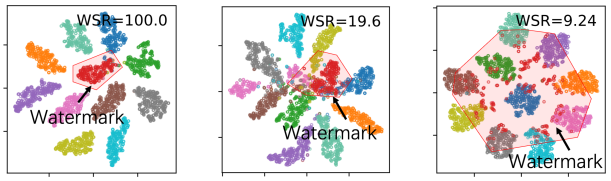
$$\mathcal{L} = \mathcal{L}_{\text{Cri}} - \lambda_1 \mathcal{L}_{\text{RepS}} + \lambda_2 \mathcal{L}_{\text{CD}^2}, \tag{7}$$

where $\mathcal{L}_{\text{Cri}}$ is the criterion loss that preserves both domain utility and watermark accuracy:

$$\mathcal{L}_{\text{Cri}} = \frac{1}{N} \sum_{\mathbf{x} \in X} \mathcal{L}(F(\mathbf{x}), F_1(\mathbf{x})) + \frac{1}{N_w} \sum_{(\mathbf{x}, y) \in D_w} \mathcal{L}(F(\mathbf{x}), y), \tag{8}$$

and $\lambda_1, \lambda_2 > 0$ are weighting factors for the auxiliary terms. The computation of $\mathcal{L}$ incurs complexity of $O(B_w^2 DK)$, where B_w is the batch size of watermark samples and D is the dimension of the L-layer output. This complexity is minimal and incurs negligible overhead, as B_w is typically much smaller than the domain batch size.

Resilience Bonus of CD^2 Optimization. Beyond improving post-MEA resilience, CD^2 optimization also enhances robustness against learning-induced removal, because the removal-post model $\hat{F}$ shares the same NTK-form approximation with the MEA-post model. As a result, minimizing $\mathcal{L}_{\text{CD}^2}$ preserves intra-class cohesion not only during extraction but also in the learning-induced removal process.



(a) CFW (b) WRK-attacked CFW (c) No watermark

Figure 3: Visualized representations of the last hidden layer.

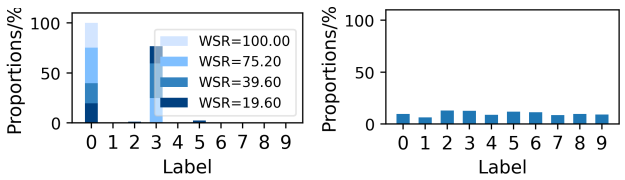
Verify CFW with Intra-class Clustering

Watermark verification is a hypothesis test with two possible outcomes: Hypothesis 0 (H0) assumes the model is watermarked, while Hypothesis 1 (H1) asserts it is not. Existing methods (Jia et al. 2021; Lv et al. 2024) typically apply t-tests and use the watermark success rate (WSR) as the test statistic (Jia et al. 2021). However, these approaches ignore the class-level structure in CFWs. Given their enhanced stability, our verification method prioritizes group clustering (Jin et al. 2023) over individual predictions. Next, we explore why clustering provides a more resilient verification evidence than prediction accuracy (*i.e.*, watermark success rate, WSR).

Clustering is a Resilient Evidence for Watermark Presence. To compare the resilience of intra-class clustering and WSR, we apply the WRK attack to class-feature watermarks. This attack perturbs the sample-wise artifacts and leverages learning-induced forgetting, making it a strong testbed for resilience.

Experiments. Experiments are conducted on CIFAR-10 with ResNet-18, using 250 out-of-domain (OOD) samples from CIFAR-20 to create the watermark task, whose clustering is optimized with CD^2. In this setting, CFW shows a weak correlation between WSR and clustering, making it ideal for our comparison. Figure 3 uses t-SNE (Van der Maaten and Hinton 2008) to visualize the final-layer representations and WSR for three models: the original watermarked model, the WRK-attacked watermarked model, and a non-watermarked model. Figure 3b shows that despite WSR dropping from 100% to 19.60%, representations remain well-clustered, indicating clustering is more resilient than WSR.

Verify with Clustering in Label-only Access. Since clustering is more resilient than WSR, we propose to verify CFW by checking for clustering. However, in practice, the



(a) WRK-Attacked CFW (b) No watermark

Figure 4: Predicted label histograms during WRK attacks.

verifier may only have **label-only** access. In this case, we observe whether label distributions imply clustering in the watermark class. Thanks to the stability enforced by CD^2 optimization, we infer that its logits are still highly consistent during removal. As a result, label distributions may reveal clustering patterns even when WSR suggests failure. This phenomenon is referred to as label clustering.

Label Clustering. We define a *deformation label* to describe new cluster centers that emerge after removal. Figure 4a shows label histograms of the watermark task during WRK attacks performed in the section titled *Experiments for WRK*. After removal, strong clustering persists on both the watermark label ($= 0$) and a deformation label ($= 3$), due to intra-class representation clustering. Thus, verification with clustering is conducted to evaluate whether the watermark and deformation labels exhibit clustering. The deformation label is not always identical but is predictable and related to the watermark label rather than the watermark samples, which are discussed in Appendix B (Xiao et al. 2025).

Experiments for Class-feature Watermarks

Overview of Experimental Setup

We evaluate five tasks spanning three domains: ResNet-18 (He et al. 2016) trained on **image** datasets (CIFAR-10, CIFAR-20 (Krizhevsky, Hinton et al. 2009)), ResNet-50 (He et al. 2016) trained on ImageNette (Howard 2019), DPCNN (Johnson and Zhang 2017) with BERT embeddings trained on a **text** dataset (DBPedia (Auer et al. 2007)), and VGG19-BN (Simonyan and Zisserman 2014) trained on an **audio** dataset (Speech Commands (Warden 2018)). The CFW dataset is built using multi-class OOD samples assigned to a single label, with size constrained to 0.2%–0.3% of the domain dataset. The watermark samples and MEA query pool are strictly disjoint, with no overlap in classes or distributions. For clustering-based verification, we introduce two metrics: intra-class variance (Var) measuring representation compactness, and label clustering (WSR_{LC}), defined as the sum of WSRs over watermark and deformation labels. Detailed CFW settings, MEA setups, and metric definitions are provided in Appendix D (Xiao et al. 2025).

Overall Evaluation of CFW

We evaluate Class-Feature Watermarks (CFW) against the six properties defined in the ownership game. Table 3 summarizes performance under WRK, focusing on utility preservation (**Prop.1**), MEA transferability (**Prop.2**), correctness (**Prop.3**), and resilience on both victim and copy models (**Prop.4**, **Prop.5**). Besides, Appendix E (Xiao et al. 2025) further evaluates: (1) resilience comparison between CFW and baseline watermarks using WSR and WSR_{LC} metrics, (2) resilience under additional removal attacks, (3) impacts of CD^2 and RepS optimization, (4) generalization across architectures, and (5) stealthiness (**Prop.6**) through anomaly detection analysis.

Results. CFW achieves near-perfect label clustering rates ($\text{WSR}_{\text{LC}} = 100\%$) on watermarked victim models, enabling high-confidence verification. The watermark transfers consistently to copy models extracted via MEA, with WSR_{LC} exceeding 81.33% before removal, which attributes to the RepS optimization. Additionally, model utility degradation remains bounded at $\leq 0.4\%$ except for ImageNette, outperforming existing black-box watermarks which cause $> 1.5\%$ degradation (Table 1). On ImageNette, CFW leads to a 2% utility drop because high-resolution inputs amplify the impact of OOD entanglement on the domain task, but its utility remains comparable to benchmarks. Lastly, CFW demonstrates exceptional resilience against removal attacks. Victim models maintain $\text{WSR}_{\text{LC}} \geq 94.67\%$ under WRK attacks, while copy models preserve $\text{WSR}_{\text{LC}} \geq 70.15\%$ despite the combined distortion of MEA and WRK. This robustness stems from two factors: CFW resists WRK's *Boundary Reshaping* by removing vulnerable decision boundaries, and withstands WRK's *Feature Shifting* with clustering-based verification, since *Feature Shifting* has a trivial effect on disrupting the representation clustering.

Evaluation on CFW Variants

We ablate the two CFW optimization components: RepS and CD^2, and compare RepS with the SNNL (Kornblith et al. 2019) algorithm for representation entanglement. The experiments are conduted on CIFAR-10 and further evaluate representation entanglement ($\mathcal{RE}$, Equation 1) and pairwise distance projections on distortion (CD^2, Equation 6) metrics. Table 4 presents quantitative results and Figure 5 shows watermark decoupling curves.

Results. RepS and SNNL both enhance $\mathcal{RE}$ (0.92 and 0.85 vs. baseline 0.19), but SNNL degrades accuracy by $> 3\%$ while delivering inferior transferability. In contrast, RepS maintains model utility ($\Delta\text{ACC} < 0.4\%$) while achieving higher $\mathcal{RE}$ and copy WSR_{LC} (94.00% vs. SNNL's 77.33%). Then, CD^2 optimization reduces CD^2 loss by $30\times$ (to 8.10×10^{-2}) and constrains copy model variance to 2.74×10^2, which is comparable to victim models (1.68×10^2) and $3.8\times$ lower than the baseline. This directly enhances WSR_{LC} from 70.00% to 94.00% in the full CFW implementation. Finally, Figure 5 demonstrates that WSR_{LC} consistently provides clearer evidence than WSR. While RepS only improves WSR resilience, it fails to maintain intra variance clustering. CD^2 optimization enables both robust clustering stability and attack resilience, with CD^2 only variants maintaining label and representation clustering even when WSR drops below 20%.

Non-watermark		Victim Model				Copy Model						
No Removal		No Removal		WRK Removal		MEA	No Removal			WRK Removal		
ACC	WSR_{LC}	ACC	WSR_{LC}	ACC	WSR_{LC}		ACC	FID	WSR_{LC}	ACC	FID	WSR_{LC}
CIFAR-10												
93.55±0.19;12.00±2.67		93.26±0.12;100.00±0		91.95±0.12;96.67±0.67		MExMI	89.29±1.12;92.70±1.06;94.00±1.33			88.94±0.95;90.13±1.12; 79.33±2.67		
						ActiveThief	85.89±1.18;88.34±1.18;88.00±0.67			87.26±1.21;89.27±1.24; 74.67±2.00		
CIFAR-20												
81.61±0.35;6.67±0.67		81.26±0.32;100.00±0		80.54±0.21;96.67±0.67		MExMI	80.64±1.24;82.35±1.36;85.33±1.33			80.18±1.27;81.97±1.42; 70.67±2.67		
						ActiveThief	71.41±1.33;77.47±1.48;81.33±1.92			72.15±1.36;77.30±1.54; 64.67±1.33		
Imagenette												
88.12±0.28;11.35±0.71		85.42±0.55;100.00±0		84.35±0.42;95.33±3.56		MExMI	86.04±1.39;83.92±0.66;86.67±2.00			84.35±0.92;82.93±0.72; 76.67±1.23		
						ActiveThief	85.32±1.14;82.51±0.78;83.33±1.33			84.49±0.88;81.92±0.84; 73.33±1.33		
DBPedia												
98.17±0.09;15.39±0.75		98.03±0.09;100.00±0		97.85±0.21;94.92±0.45		MExMI	95.15±0.71;96.62±0.96;97.33±0.38			94.83±0.54;95.76±1.02; 88.72±0.42		
						ActiveThief	91.35±0.87;92.82±1.08;94.32±0.57			91.51±0.63;93.05±1.14; 80.64±0.61		
Speech Commands												
97.36±0.12;8.66±0.47		97.02±0.14;100.00±0		96.03±0.24;95.33±0.27		MExMI	96.46±0.66;97.82±1.26;95.00±0.73			95.96±0.69;96.79±1.32; 82.12±0.87		
						ActiveThief	96.33±0.72;97.73±1.38;94.00±0.99			94.61±0.75;95.20±1.44; 79.35±1.11		

Table 3: Performance of Class-feature Watermark (CFW) Against WRK Removal

Watermarks	Victim Model						Copy Model				
Metrics	$\mathcal{RE}\uparrow$	$CD^2\downarrow$	ACC	WSR	WSR_{LC}	Var($\times 10^2$)	ACC	FID	WSR	WSR_{LC}	Var($\times 10^2$)
w/o CD^2, RepS	0.19±0.06	3.98±0.25	93.70±0.15	100.00±0.0	100.00±0.0	2.59±0.44	89.55±1.23	92.54±1.13	57.33±2.00	70.00±2.67	10.30±3.25
w/ SNNL only	0.85±0.05	2.51±0.18	90.13±0.22	99.33±0.15	100.00±0.0	4.14±1.21	87.18±1.14	90.49±0.97	73.33±2.00	77.33±2.67	11.12±3.78
w/ RepS only	0.92±0.04	1.78±0.15	93.31±0.12	100.00±0.0	100.00±0.0	1.92±0.64	89.85±1.18	92.85±1.28	92.00±1.33	94.00±1.33	8.17±2.84
w/ CD^2 only	0.04±0.03	0.059±0.008	93.40±0.14	100.00±0.0	100.00±0.0	1.35±1.02	89.85±1.21	92.11±1.32	68.67±2.00	84.00±2.67	2.47±1.01
CFW	0.75±0.06	0.081±0.010	93.26±0.12	100.00±0.0	100.00±0.0	1.68±0.52	89.29±1.12	92.70±1.06	91.33±1.33	94.00±1.33	2.74±0.90

Arrows represent the trend toward better watermark performance.

Table 4: Evaluation Results of Class-feature Watermark (CFW) Variants

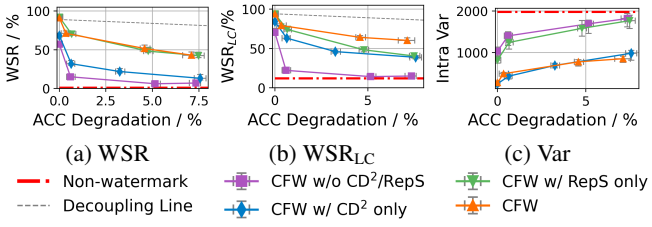


Figure 5: Watermark decoupling curves for CFW on extracted copy models. Vertical lines show the error bars. Appendix E.3 presents corresponding curves on victim models.

Related Works

Black-box Watermarks and Their Resilience to Removal. Model watermarking is a promising technique for verifying ownership of machine learning models. While early work showed MEAs could strip watermarks (Lukas et al. 2022), recent breakthroughs demonstrate that black-box watermarks can survive extraction attacks by leveraging representation entanglement (RE) (Jia et al. 2021; Lv et al. 2024). However, their resilience against consecutive MEA and watermark removal attacks remains unevaluated.

Existing removal methods exploit three decoupling strategies: **reversion-based removal**, which reconstructs and erases watermark triggers (NC (Wang et al. 2019), I-BAU (Zeng et al. 2022), DTI-DBF (Xu et al. 2023), Aiken (Aiken et al. 2021)); **neuron pruning**, which eliminates neurons associated with watermark tasks (FP (Liu, Dolan-Gavitt, and Garg 2018), CLP (Zheng et al. 2022b)); and **learning-induced forgetting**, which removes non-domain features via fine-tuning (NAD (Li et al. 2021), SEAM (Zhu et al. 2023), FST (Min et al. 2023)). Watermarks with weak entanglement, such as RS (Bansal et al. 2022) and MBW (Lukas et al. 2022), are vulnerable to all three strategies. Strongly entangled watermarks like EWE (Jia et al. 2021) and MEA-D (Lv et al. 2024) resist most removal attacks but remain vulnerable to DTI-DBF and FST. On the other hand, the two removal methods show limited effectiveness against entangled Blend-type watermarks.

Conclusion

This paper first exposes resilience gaps in black-box watermarks against model extraction attacks (MEA). While existing removal methods are often constrained by the representation entanglement deliberately designed in these watermarks, our Watermark Removal attacK (WRK) overcomes this entanglement barrier by exploiting decision boundaries shaped by their sample-level artifacts. To address the vulnerability exposed by WRK, we introduce Class-Feature Watermarks (CFWs) which shift to class-level artifacts for enhanced resilience and further optimize representation entanglement and stability during MEA. CFW achieves watermark success rates of $\geq 70.15\%$ under combined MEA and WRK attacks across domains, significantly outperforming prior methods in resilience.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No: 92270123), and the Research Grants Council (Grant No: 15209922, 15210023, 15224124 and 15207725), Hong Kong SAR, China.

References

Aiken, W.; Kim, H.; Woo, S.; and Ryoo, J. 2021. Neural network laundering: Removing black-box backdoor watermarks from deep neural networks. *Computers & Security*, 106: 102277.

Alexey, K.; Ian, G.; and Samy, B. 2016. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*.

Alharbi, S.; Alrazgan, M.; Alrashed, A.; Alnomasi, T.; Almojel, R.; Alharbi, R.; Alharbi, S.; Alturki, S.; Alshehri, F.; and Almojil, M. 2021. Automatic speech recognition: Systematic literature review. *Ieee Access*, 9: 131858–131876.

Amazon Web Services. 2017. Amazon SageMaker. https://aws.amazon.com/sagemaker/, Accessed: 2024-12-22.

Auer, S.; Bizer, C.; Kobilarov, G.; Lehmann, J.; Cyganiak, R.; and Ives, Z. 2007. Dbpedia: A nucleus for a web of open data. In *international semantic web conference*, 722–735. Springer.

Bansal, A.; Chiang, P.-y.; Curry, M. J.; Jain, R.; Wigington, C.; Manjunatha, V.; Dickerson, J. P.; and Goldstein, T. 2022. Certified neural network watermarks with randomized smoothing. In *International Conference on Machine Learning*, 1450–1465. PMLR.

Bennani, M. A.; Doan, T.; and Sugiyama, M. 2020. Generalisation guarantees for continual learning with orthogonal gradient descent. *arXiv preprint arXiv:2006.11942*.

Bottou, L. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT*, 177–186. Springer.

Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; and Sander, J. 2000. LOF: Identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, 93–104. ACM.

Chen, R.; Duffy, Á.; Petrazzini, B. O.; Vy, H. M.; Stein, D.; Mort, M.; Park, J. K.; Schlessinger, A.; Itan, Y.; Cooper, D. N.; et al. 2024. Expanding drug targets for 112 chronic diseases using a machine learning-assisted genetic priority score. *Nature Communications*, 15(1): 8891.

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

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Doan, T.; Bennani, M. A.; Mazoure, B.; Rabusseau, G.; and Alquier, P. 2021. A Theoretical Analysis of Catastrophic Forgetting through the NTK Overlap Matrix. In Banerjee, A.; and Fukumizu, K., eds., *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, 1072–1080. PMLR.

Google Cloud. 2024. Google AutoML. https://cloud.google.com/automl, Accessed: 2024-12-22.

Graves, L.; Nagisetty, V.; and Ganesh, V. 2021. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 11516–11524.

Gu, T.; Liu, K.; Dolan-Gavitt, B.; and Garg, S. 2019. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Howard, J. 2019. Imagenette. https://github.com/fastai/imagenette, Accessed: 2025-04-19.

Jia, H.; Choquette-Choo, C. A.; Chandrasekaran, V.; and Papernot, N. 2021. Entangled watermarks as a defense against model extraction. In *30th USENIX security symposium (USENIX Security 21)*, 1937–1954.

Jin, Y.; Zhang, X.; Lou, J.; Ma, X.; Wang, Z.; and Chen, X. 2023. Explaining adversarial robustness of neural networks from clustering effect perspective. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4522–4531.

Johnson, R.; and Zhang, T. 2017. Deep pyramid convolutional neural networks for text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 562–570.

Kim, B.; Lee, S.; Lee, S.; Son, S.; and Hwang, S. J. 2023. Margin-based neural network watermarking. In *International Conference on Machine Learning*, 16696–16711. PMLR.

Kornblith, S.; Norouzi, M.; Lee, H.; and Hinton, G. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, 3519–3529. PMLR.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Li, Y.; Lyu, X.; Koren, N.; Lyu, L.; Li, B.; and Ma, X. 2021. Neural Attention Distillation: Erasing Backdoor Triggers from Deep Neural Networks. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Li, Y.; Lyu, X.; Ma, X.; Koren, N.; Lyu, L.; Li, B.; and Jiang, Y.-G. 2023. Reconstructive neuron pruning for backdoor defense. In *International Conference on Machine Learning*, 19837–19854. PMLR.

Liang, Z.; Ye, Q.; Wang, Y.; Zhang, S.; Xiao, Y.; Li, R.; Xu, J.; and Hu, H. 2025. "Yes, My LoRD." Guiding Language Model Extraction with Locality Reinforced Distillation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria*. Association for Computational Linguistics.

Lin, J.; Xu, L.; Liu, Y.; and Zhang, X. 2020. 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*, 113–131.

Liu, F. T.; Ting, K. M.; and Zhou, Z.-H. 2008. Isolation Forest. In *2008 Eighth IEEE International Conference on Data Mining*, 413–422. IEEE.

Liu, J.; Zhang, R.; Szyller, S.; Ren, K.; and Asokan, N. 2024. False claims against model ownership resolution. In *33rd USENIX Security Symposium (USENIX Security 24)*, 6885–6902.

Liu, K.; Dolan-Gavitt, B.; and Garg, S. 2018. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, 273–294. Springer.

Lukas, N.; Jiang, E.; Li, X.; and Kerschbaum, F. 2022. SoK: How Robust is Image Classification Deep Neural Network Watermarking? In *43rd IEEE Symposium on Security and Privacy, SP 2022, San Francisco, CA, USA, May 22-26, 2022*, 787–804. IEEE.

Lv, P.; Ma, H.; Chen, K.; Zhou, J.; Zhang, S.; Liang, R.; Zhu, S.; Li, P.; and Zhang, Y. 2024. MEA-Defender: A Robust Watermark against Model Extraction Attack. *2024 IEEE Symposium on Security and Privacy (SP)*.

Microsoft Azure. 2024. Azure Machine Learning by Microsoft. https://azure.microsoft.com/en-us/products/machine-learning/, Accessed: 2024-12-22.

Min, R.; Qin, Z.; Shen, L.; and Cheng, M. 2023. Towards stable backdoor purification through feature shift tuning. *Advances in Neural Information Processing Systems*, 36: 75286–75306.

Nguyen, T. A.; and Tran, A. T. 2021. WaNet - Imperceptible Warping-based Backdoor Attack. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Pal, S.; Gupta, Y.; Shukla, A.; Kanade, A.; Shevade, S.; and Ganapathy, V. 2020. Activethief: Model extraction using active learning and unannotated public data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 865–872.

Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; and Chen, L.-C. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4510–4520.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Tramèr, F.; Zhang, F.; Juels, A.; Reiter, M. K.; and Ristenpart, T. 2016. Stealing machine learning models via prediction {APIs}. In *25th USENIX security symposium (USENIX Security 16)*, 601–618.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Wang, B.; Yao, Y.; Shan, S.; Li, H.; Viswanath, B.; Zheng, H.; and Zhao, B. Y. 2019. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE symposium on security and privacy (SP)*, 707–723. IEEE.

Warden, P. 2018. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*.

Xiao, Y.; Ye, Q.; Hu, H.; Zheng, H.; Fang, C.; and Shi, J. 2022. MExMI: Pool-based active model extraction crossover membership inference. *Advances in Neural Information Processing Systems*, 35: 10203–10216.

Xiao, Y.; Ye, Q.; Liang, Z.; Li, H.; Li, R.; Zheng, H.; and Hu, H. 2025. Class-feature Watermark: A Resilient Black-box Watermark Against Model Extraction Attacks. *arXiv preprint arXiv:2511.07947*.

Xu, X.; Huang, K.; Li, Y.; Qin, Z.; and Ren, K. 2023. Towards reliable and efficient backdoor trigger inversion via decoupling benign features. In *The Twelfth International Conference on Learning Representations*.

Zeng, Y.; Chen, S.; Park, W.; Mao, Z.; Jin, M.; and Jia, R. 2022. Adversarial Unlearning of Backdoors via Implicit Hypergradient. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Zhang, K.; Cheng, S.; Shen, G.; Tao, G.; An, S.; Makur, A.; Ma, S.; and Zhang, X. 2024. Exploring the Orthogonality and Linearity of Backdoor Attacks. In *2024 IEEE Symposium on Security and Privacy (SP)*, 225–225. IEEE Computer Society.

Zhang, X.; Hu, H.; Ye, Q.; Bai, L.; and Zheng, H. 2025. MER-Inspector: Assessing Model Extraction Risks from An Attack-Agnostic Perspective. In *Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia*, 4300–4315. ACM.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Zheng, H.; Ye, Q.; Hu, H.; Fang, C.; and Shi, J. 2019. BDPL: A Boundary Differentially Private Layer Against Machine Learning Model Extraction Attacks. In Sako, K.; Schneider, S. A.; and Ryan, P. Y. A., eds., *Computer Security - ESORICS 2019, Part I*, volume 11735 of *Lecture Notes in Computer Science*, 66–83. Springer.

Zheng, H.; Ye, Q.; Hu, H.; Fang, C.; and Shi, J. 2022a. Protecting Decision Boundary of Machine Learning Model With Differentially Private Perturbation. *IEEE Trans. Dependable Secur. Comput.*, 19(3): 2007–2022.

Zheng, R.; Tang, R.; Li, J.; and Liu, L. 2022b. Data-free backdoor removal based on channel lipschitzness. In *European Conference on Computer Vision*, 175–191. Springer.

Zhu, H.; Liang, S.; Hu, W.; Li, F.; Jia, J.; and Wang, S. 2024. Reliable Model Watermarking: Defending Against Theft without Compromising on Evasion. *arXiv preprint arXiv:2404.13518*.

Zhu, R.; Tang, D.; Tang, S.; Wang, X.; and Tang, H. 2023. Selective amnesia: On efficient, high-fidelity and blind suppression of backdoor effects in trojaned machine learning models. In *2023 IEEE Symposium on Security and Privacy (SP)*, 1–19. IEEE.