SEAL: Entangled White-box Watermarks on Low-Rank Adaptation

Anonymous ACL submission

Abstract

Watermarking is a promising copyright protection method for Deep Neural Networks (DNNs). It works by embedding a secret identity message into the DNN during training, and extracting it later when copyright is disputed. Prior work has proposed various techniques that can embed secret identity messages into different layers of a DNN. We observe that models nowadays are frequently created and distributed in the form of Low-Rank Adaptation (LoRA) weights, because of its significant savings in training cost. We propose **SEAL** (SEcure wAtermarking on LoRA weights), the first watermarking method tailored for LoRA weights. Unlike existing methods that focus on specific layers and are unsuitable for LoRA's unique structure, SEAL embeds a secret, non-trainable matrix between trainable LoRA weights, serving as a passport to claim ownership. SEAL then entangles this passport with the LoRA weights through finetuning, and distributes the finetuned weights after hiding the passport. We demonstrate that SEAL is robust against a variety of known attacks, and works without compromising the performance of watermarked models on various NLP tasks.

1 Introduction

Recent years have witnessed an increasing demand for protecting deep neural networks (DNNs) as intellectual properties (IPs), mainly due to the significant cost of collecting quality data and training DNNs on it. In response, researchers have proposed various DNN watermarking methods for DNN copyright protection (Uchida et al., 2017; Darvish Rouhani et al., 2019; Zhang et al., 2018; Fan et al., 2019; Zhang et al., 2020; Xu et al., 2024; Lim et al., 2022), which work by secretly embedding identity messages into the DNNs during training. The IP holders can present the identity messages to a verifier in the event of a copyright dispute to claim ownership.

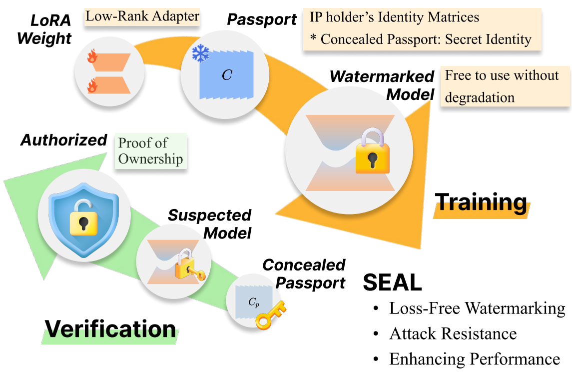


Figure 1: SEAL scheme: A passport matrix C is embedded into LoRA weights during training, creating a watermarked model. The concealed passport C_p verifies ownership, ensuring loss-free watermarking, attack resistance, and performance enhancement.

Meanwhile, recent advances in Parameter-Efficient FineTuning (PEFT), particularly Low-Rank Adaptation (LoRA) (Hu et al., 2022), have been transforming the way the majority of domain-specific DNNs are built. LoRA is the *de facto* method and format in the open-source community because of its properties—light-weight, no inference latency, and offers performance comparable to full finetuning. Although LoRA utilizes pretrained foundation models, the finetuning results reside entirely within the LoRA adapters, which should be considered IPs. As Luo et al., 2024 has reported, over 100K LoRA weights are shared on platforms like Civit AI[1], indicating their high prevalence. Unfortunately, existing white-box DNN watermarking schemes are not suitable for LoRA where weights are released in open source, as they only support embedding identity messages in specific architecture-bounded component, such as kernels in convolutional layer (Uchida et al., 2017; Liu et al., 2021; Zhang et al., 2020; Lim et al., 2022). These methods are not suitable for the unique requirements of LoRA, highlighting the

[1] https://civitai.com

need for a specialized watermarking solution.

This paper proposes SEAL, the first watermarking scheme designed to protect the copyright of LoRA weights. The key idea of SEAL is to integrate a constant matrix within the LoRA framework, acting as a hidden identity message that is difficult to extract, remove, modify or even counterfeit, thus offering robust IP protection. A constant in SEAL, non-trainable matrix, which is entangled with the up and down blocks of LoRA. This constant matrix in SEAL naturally directs the gradients through itself during finetuning, eliminating the need to design additional constraint losses for watermark embedding. Additionally, after training ends, SEAL decomposes the constant matrix into two and integrates each into the up and down blocks of LoRA, respectively. This decomposition ensures that the resulting model appears indistinguishable from a standard LoRA-trained model to external observers, offering a versatile and less intrusive method for safeguarding DNNs.

We validate the robustness of SEAL as an IP protection mechanism with a variety of concrete attacks reported in the literature, namely removal (See et al., 2016), obfuscation (Yan et al., 2023; Pegoraro et al., 2024), and ambiguity attacks (Fan et al., 2019). To successfully remove identity messages, we show in Section 4.6 that an attacker would need to zero out **99.9%** of the weights, which in turn results in severe performance degradation of the host task. In Section 4.6, we demonstrate that SEAL is structurally immune to the structural obfuscation attack recently proposed by (Yan et al., 2023).We additionally show in Section 4.7 that an adversary would need to generate a matrix with over 70% similarity to the hidden passport to pass the verification process, thus demonstrating SEAL's robustness against ambiguity attacks.

Importantly, SEAL's robustness against these attacks comes at virtually no fidelity cost; applying SEAL does not degrade the performance of the original task. Our fidelity evaluation shows that SEAL achieves performance comparable to, and sometimes even surpassing, standard LoRA in tasks ranging from commonsense reasoning to instruction tuning.

Our contributions are three-fold:

1. **Simple yet Strong Copyright Protection for LoRA:** We present SEAL, the first watermarking scheme for protecting LoRA weights by embedding a hidden identity message using a constant matrix, eliminating the need for additional loss terms, offering a straightforward yet robust solution.
2. **Robustness Against Attacks:** We demonstrate SEAL's resilience against various attacks, including removal, obfuscation, and ambiguity attacks, showing it maintains robust IP protection even under severe adversarial conditions.
3. **Enhanced Performance:** Our approach ensures structural camouflage and functional invariance, meaning that applying SEAL does not degrade the performance of the task. In fact, our fidelity evaluation indicates that SEAL achieves performance comparable to, and sometimes even surpassing.

2 Preliminary

2.1 Low-Rank Adaptation

LoRA (Hu et al., 2022) is an adaptation method based on the premise that specific tasks has "intrinsic low rank" within the full parameter space of a model. LoRA leverages the capabilities of a pretrained model, transferring its performance on a specific task. During training, the pretrained model's weights, $W \in \mathbb{R}^{b\times a}$, remain frozen, and only two low-rank decomposed matrices, $A \in \mathbb{R}^{r\times a}$ and $B \in \mathbb{R}^{b\times r}$, are treated as trainable parameters.

$$W^{'} = W + \Delta W = W + BA \tag{1}$$

The absence of activation functions between A and B allows for efficient integration into the pretrained model after training by simply adding BA to the original weights.

2.2 White-box Watermarks

We focus on white-box scenarios where model weights are publicly accessible. This setup is natural for LoRAs, as their entire weights are usually shared due to their smaller size compared to full models (Hu et al., 2022).

Existing white-box watermarking methods can be broadly categorized into three types based on where the secret message is embedded (Yan et al., 2023): weight-, activation-, and passport-based.

- *Weight-based* methods embed watermarks, a secret bit sequence consisting of values such as {1, -1}, directly into the model weights. (Uchida et al., 2017, Liu et al., 2021, Fernandez et al., 2024)

- *Activation-based* methods utilize activation maps for special input and layer pair to embed the identity messages of the IP holder (Darvish Rouhani et al., 2019, Lim et al., 2022).
- *Passport-based* methods, first introduced by Fan et al., 2019, adds a so-called passport layer, a linear layer with scale factors and bias shifts following a convolutional layer. This passport layer embeds a unique identifier, *passport*, into the neural network. During verification, a forged passport can be detected because the model's performance degrades with invalid passports. Zhang et al., 2020 extended this concept to normalization layers.

2.3 Attacks on Watermarks

Attacks on white-box DNN watermarks are categorized into three types: removal, obfuscation, and ambiguity attacks. Table 1 shows that what are the targets of each attack method.

Attack	Target	
	Identity	Verification
Removal	Erase	Invalidate
Obfuscation	Disregard	Invalidate
Ambiguity	Forge	Bypass

Table 1: Attack and its purpose on each target type

Removal/Obfuscation Attacks aim to remove or obfuscate the identity messages embedded in the models such that the original identity information cannot be extracted in the verification phase. We show that SEAL has robustness against removal/obfuscation attacks in Section 4.6.

- *Pruning:* This attack involves eliminating neurons that are deemed unnecessary or have minimal impact on the DNN's inference process (Uchida et al., 2017; Darvish Rouhani et al., 2019). It is straight way to remove embedded identity. Usually, pruning attacks zeroing out model's weight based on its L1-norms.
- *Fine-tuning:* If the dataset used to train the DNN is publicly accessible, attackers can retrain the victim model without the watermark constraint loss (Chen et al., 2021; Guo et al., 2021; Yan et al., 2023).
- *Structural Obfuscation:* (Yan et al., 2023; Pegoraro et al., 2024) recently proposed attack method focuses solely on disrupting the watermark verification process with modifying the structure of the DNN, while preserving its original functionality. When verification process launched, verifier cannot retrieve watermark from obfuscated structure of weight because distribution of its parameter has been changed.

Ambiguity Attacks aim to falsely claim ownership by forging counterfeit watermarks. The adversaries can deceive the verifier into recognizing them as the rightful owner (Fan et al., 2019; Zhang et al., 2020; Chen et al., 2023). Each DNN watermarking scheme needs specific countermeasures to address ambiguity attacks effectively. For instance, Chen et al., 2023 train an additional layer to replace the passport, acting as a counterfeit watermark.

2.4 Criteria for Evaluation

Measure of Success. A defensive algorithm for attacks on DNN watermarks must satisfy the following requirements (Uchida et al., 2017):

- *Fidelity:* The insertion of a watermark should not degrade the performance of the host task. If any performance degradation occurs, it should be minimal or justified by a trade-off with other benefits.
- *Robustness:* Once embedded, the watermark should be resistant to attempts to remove or obfuscate the identity messages. If an attacker manages to remove or obfuscate them, it should come at a significant degradation of the host task's performance, or a computational cost comparable to the original finetuning cost.

Attacker. We consider an adversary who attempts to attack open-sourced watermarked LoRA weights for a known base model. The goal of the adversary is to nullify the ownership verification of the LoRA weights, either by extracting the watermark, by erasing it, or by embedding their own, counterfeit watermark over the original one. We assume that the adversary has the following capabilities:

- *Minimal Utility Loss:* The adversary should not undermine the utility of the model. Otherwise such attack is futile as the attacker cannot benefit from a malfunctioning model.
- *Limited Computational Cost:* Compromising the watermark should not require computational resources larger than those required for training LoRA weights by adversaries themselves.
- *No Dataset Access:* As many LoRA training processes involve proprietary assets of the model owners, access to the original training data cannot always be taken for granted. Thus, the adversary's goal should be to undermine the owner's watermarks without access to the original train-

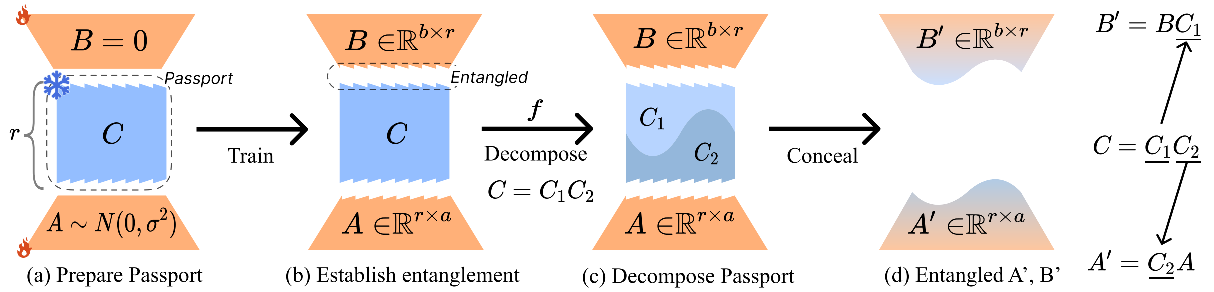


Figure 2: SEAL Scheme: The figure illustrates the overall process of the SEAL watermarking method. **(a)** A constant matrix C is initialized along with LoRA weights A and B. **(b)** During training, C is entangled with the LoRA weights. **(c)** After training, C is decomposed into C_1 and C_2. **(d)** The decomposed parts are concealed within the weights, resulting in entangled weights A' and B'. Detailed forward and backward passes are in Appendix B.

ing data. Otherwise, the adversary can build their own model from scratch, eliminating the need for an attack in the first place.

- *Watermark Knowledge:* Based on Kerckhoff's principle, we assume that the adversary knows about SEAL but does not know the exact watermark embedded.

3 SEAL: The Watermarking Scheme

Previous methods (Fan et al., 2019; Zhang et al., 2020) are architecture-dependent, and while our approach is also dependent on LoRA, direct comparisons are challenging. However, due to similarities with passport-based watermarking—such as using a linear layer, embedding the watermark within the passport layer, concealing the passport, and used during training—we categorize our method as passport-based.

As depicted in Figure 2, SEAL at a high level operates as follows. First, SEAL introduces the given passport as a non-trainable matrix, in between the training of trainable parameters B and A. Next, we train the weights on the host task with this non-trainable passport present. Once trained, SEAL decomposes the passport into two, which are then concealed by multiplying each with B and A, respectively. The final results, denoted by B' and A' are distributed as LoRA weights.

Throughout this section, we use notations introduced by Fan et al., 2019, with additional definitions and adaptations as necessary. We summarized them in Appendix A.

3.1 Entangling Passports during Training

SEAL embeds the watermark during training by inserting the non-trainable, constant matrix C between the trainable parameters A and B. Doing so effectively *entangles* the given passport with A and B. The concept of entanglement is superficially similar to the entanglement proposed by Jia et al., 2021. It involves indistinguishable distributions between host and watermarked tasks. In our context, we define entanglement as follows.

Definition 1 (Entanglement)**.** Given trainable parameters A and B, and a non-trainable parameter C, A and B are in *entanglement* via C if and only if they produce the correct output only when C is present between them.

Another difference between SEAL and prior work is that SEAL eliminates the need for additional loss functions to embed the watermark. C directly influences the computations of A and B during the forward pass, and modifies the gradient flow in the backward pass, thereby embedding itself through normal training process. Details of training both passes can be found in Appendix B.

3.2 Hiding Passports for Distribution

After successfully establishing the entanglement between the passport and other trainable parameters, the passport must be concealed before distribution. Therefore, we decompose the passport C of the IP holder into two matrices such that their product reconstructs C, as shown in Figure 2 (c). By distributing each of the the decomposed passport into trainable parameters, IP holder can hide secret passport, C.

Definition 2 (Decomposition Function)**.** For a given constant C, a function f is a decomposition function of C where $f(C) = C_1C_2$ and $C_1C_2 = C$.

An example of a watermark decomposition using SVD is

$$f_{svd}(C) = (U_C\sqrt{\Sigma_C})(\sqrt{\Sigma_C}V_C^T) \qquad (2)$$

where $U_C \Sigma_C V_C^T = C$. Using this example, the resulting matrices are

$$B' = B\,(U_C\sqrt{\Sigma_C}) \text{ and } A' = (\sqrt{\Sigma_C}V_C^T)\,A \quad (3)$$

This process ensures that models trained with SEAL, which contain three matrices per layer, $\mathbb{N}(A, B, C)$, can be distributed in a form that resembles standard LoRA implementations with only two matrices, $\mathbb{N}(A', B')$.

3.3 Passport-based Ownership Verification

The key idea of passport-based watermarking is that, when presented with forged passports under ambiguity attacks, the model's performance deteriorates due to which the ownership verification fails (Fan et al., 2019).

Definition 3 (Verification Process)**.** The DNN ownership verification process of SEAL, denoted by V, is defined as a three-tuple, $V(\mathbb{N}(A, B, C_t), M_t, \epsilon_V)$.

The outcome of the verification process depends on the presented passport C_t, where C_t is the runtime passport used during inference. This dependency indicates that the integrity of the verification process relies significantly on the accuracy and authenticity of the presented passport. The threshold of the verification is defined as $\epsilon_V = |M(\mathbb{N}(A, B, C)) - M_p(\mathbb{N}(A, B, C_p)|$ where C is the distributed passport and C_p is the concealed passport. With a forged passport $C_{adv} \neq C_p$, the fidelity score, $M_{adv}(\mathbb{N}(A, B, C_{adv}))$, will deteriorate such that the discrepancy is larger than a threshold, i.e., $|M_p - M_{adv}| > \epsilon_V$. This condition tests the robustness of the model against verification attempts with forged passports.

The reason why the IP holder can pass the verification process while the adversary cannot is as follows: During the verification process, the fidelity score is measured using the passport C_t submitted by either the IP holder or the adversary. To pass the verification process, C_t must be entangled with the parameters A and B. This entanglement can only occur if C_t was used during the training process. Therefore, the legitimate IP holder, who has used the passport during training, can submit C_t and pass the verification process.

Additionally, the method for extracting the passport involves multiplying $\mathbb{N}(A', B')$ with the pseudo-inverse of A and B. This allows us to retrieve the embedded passport, C from $\mathbb{N}(A', B')$.

If the adversary creates a forged triplet such that $\mathbb{N}(A', B') = \mathbb{N}(A_{adv}, B_{adv}, C_{adv})$, they still cannot create another $C_{adv'}$ with $|M_{adv'} - M_p| < \epsilon_V$. This is because the adversary does not participate in the training phase and therefore cannot acquire multiple forged passports. As a result, the nature of the entanglement process prevents the adversary from successfully passing the verification with a forged passport.

4 Experiments

4.1 Experimental Setup

Fidelity. To demonstrate that the performance of models after embedding SEAL passports does not degrade, we conducted a variety of tasks encompassing both language and image modalities. Initially, we evaluate our model by comparing it with various open-source large language models such as LLaMA-2-7B/13B (Touvron et al., 2023), LLaMA-3-8B (AI@Meta, 2024), Gemma-2B (Team et al., 2024), and Mistral-7B-v0.1 (Jiang et al., 2023) on commonsense reasoning tasks. Next, we verify the model's effectiveness on instruction tuning tasks. Following this, we extend our approach to multimodal Vision Language Model (Liu et al., 2024) by evaluating the model's performance on visual instruction tuning. Finally, we assess SEAL's capabilities on image-generative tasks (Rombach et al., 2022).

Robustness. We evaluated the robustness of SEAL against removal and ambiguity attacks by measuring the fidelity scores in commonsense reasoning tasks. For removal attacks, we verified the presence of the extracted watermark. For ambiguity attacks, we measured fidelity scores to ensure accurate verification of genuine versus counterfeit passports.

4.2 Commonsense Reasoning

Table 2 displays the comparative performance of commonsense reasoning tasks across various models, including LLaMA-2-7B/13B, LLaMA-3-8B, Gemma-2B, and Mistral-7B-v0.1. The experimental results emphasize that SEAL can be seamlessly integrated into existing LoRA architectures, making it an invaluable tool for safeguarding intellectual property without affecting the model's operational performance.

4.3 Instruction Tuning

Table 3 shows the scores for LLaMA-2-7B and Gemma-2B, instruction tuned with both LoRA and

	Method	BoolQ	PIQA	SIQA	HellaSwag	Wino	ARC-e	ARC-c	OBQA	Avg. ↑
LLaMA-2-7B	LoRA	74.56	83.41	79.89	89.06	84.61	86.95	75.51	86.80	82.60
	SEAL (Ours)	73.15	76.61	80.86	83.80	86.03	81.39	67.15	84.20	79.15
	SEAL† (Ours)	73.00	86.24	81.78	90.92	86.50	88.59	75.17	86.00	**83.53**
LLaMA-2-13B	LoRA	75.08	87.21	82.09	92.05	88.40	90.57	77.82	86.00	84.90
	SEAL (Ours)	75.32	87.27	83.52	93.83	88.95	90.49	79.95	88.60	85.99
	SEAL† (Ours)	75.32	88.90	83.42	93.91	89.42	91.33	81.40	88.20	**86.49**
LLaMA-3-8B	LoRA	73.58	86.13	80.35	91.85	85.95	90.11	78.58	85.00	83.94
	SEAL (Ours)	73.91	88.41	82.81	94.65	88.00	91.84	82.42	85.60	85.96
	SEAL† (Ours)	75.63	90.21	83.47	96.00	90.21	92.97	84.98	91.20	**88.08**
Gemma-2B	LoRA	65.96	78.62	75.23	79.20	76.64	79.13	62.80	72.40	73.75
	SEAL (Ours)	66.45	82.16	78.20	83.72	79.95	82.62	68.09	79.40	77.57
	SEAL† (Ours)	66.54	82.70	79.53	87.70	80.58	84.01	69.63	79.80	**78.81**
Mistral-7B-v0.1	LoRA	75.87	91.13	81.99	94.54	88.56	93.14	83.02	89.00	87.16
	SEAL (Ours)	73.79	86.84	81.62	90.80	87.68	90.27	79.52	88.20	84.84
	SEAL† (Ours)	77.19	90.32	82.86	94.56	89.74	93.14	83.70	91.20	**87.84**

Table 2: Accuracy comparison of eight sub-tasks of commonsense reasoning for LLaMA-2-7B/13B (Touvron et al., 2023), LLaMA-3-8B (AI@Meta, 2024), Gemma-2B (Team et al., 2024), and Mistral-7B-v0.1 (Jiang et al., 2023) using LoRA, and SEAL methods. The dataset was obtained from (Hu et al., 2023) and the hyperparameters were modified accordingly. Note: SEAL†represents a constant matrix C that was randomly initialized from a normal distribution.

Task	Inst. Tune		Text-to-Image		
	Textual	Visual			
Metric	MT-B	Avg.	CLIP-T	CLIP-I	DINO.
LoRA	5.38	66.9	0.198	0.801	0.669
SEAL	5.50	63.1	0.202	0.804	0.647

Table 3: Fidelity on wide range of Tasks. **Inst. Tune**: Instruction Tuning. **MT-B**: MT-Bench, (Zheng et al., 2023), Score of Visual Inst. Tune: average of seven vision-language tasks. CLIP-I, DINO. show subject fidelity score and CLIP-T represents prompt fidelity score.

SEAL, using a 10K subset of Alpaca dataset (Taori et al., 2023). The scores represent the average ratings given by GPT-4 on a scale of 1 to 10 for the models' responses to questions from MT-Bench (Zheng et al., 2023). Since the Alpaca dataset is optimized for single-turn interactions, the average score for single-turn performance from MT-Bench is used. The results demonstrates that applying SEAL results in no quality degradation when compared to LoRA, confirming the fidelity of SEAL.

4.4 Visual Instruction Tuning

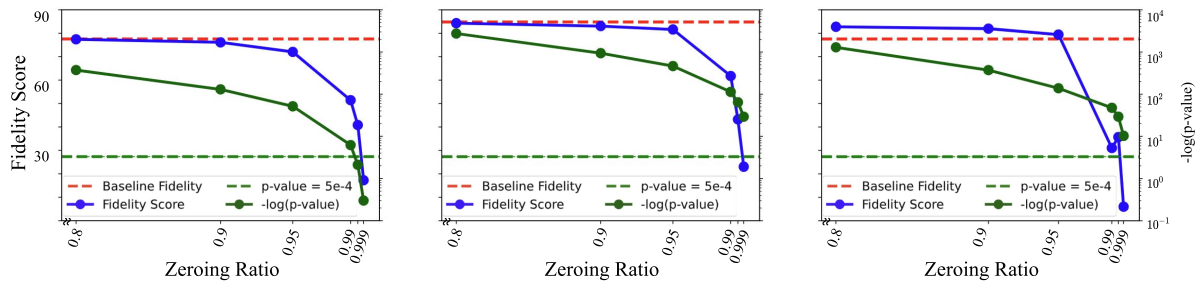
Table 3 shows the average performance across 7 visual instruction tuning benchmarks for LoRA and SEAL on LLaVA-1.5 with detailed elaboration in Appendix E. Our results indicates comparable performance between the two methods.

4.5 Text-to-Image Synthesis

The experimentation with the Stable Diffusion model (Rombach et al., 2022) in conjunction with dataset of DreamBooth (Ruiz et al., 2023) trained with LoRA elucidates the versatility and robustness of SEAL when integrated into diverse architectures. Referring to Table 3, which contains the metrics used for evaluation, we observe a detailed comparison of subject fidelity (DINO, CLIP-I) and prompt fidelity (CLIP-T). We provide detailed information of dataset, hyperparameters, and evaluation metrics on Appendix D. Our results corroborate these findings, demonstrating that SEAL can maintain high fidelity in both subject representation and prompt accuracy without degrading model performance. Additionally, comparison images of LoRA and SEAL on the same subject of the DreamBooth dataset provide visual evidence of these performance metrics; these images are available in Figure 7 .

4.6 Robustness against Removal & Obfuscation Attacks

Pruning Attacks. We conducted pruning attacks on SEAL-trained weights, $\mathbb{N}(\cdot, C)$, by zeroing out $\mathbb{N}(\cdot, C)$ based on its L1-norms. We used statistical testing instead of Bit Error Rate (BER) because,



(a) Gemma-2B (b) Mistral-7B-v0.1 (c) LLaMA-2-7B

Figure 3: Pruning Attack: The x-axis represents the zeroing ratio, the left y-axis shows the fidelity score, and the right y-axis displays the -log(p-value) on a log scale. If -log(p-value) is *above* 3.3 (i.e., p-value < 5e-4), detecting the watermark succeeds. The graphs show that as the zeroing ratio increases, the fidelity score decreases, and the -log(p-value) also decreases. This indicates the watermark remains detectable until 99.9% of the weights are zeroed, which significantly degrades the host task's performance, demonstrating SEAL's robustness against pruning attacks.

unlike prior work (Uchida et al., 2017; Fernandez et al., 2024; Zhang et al., 2020) that used a small number of bits, $N{\sim}10^2$, the amount of our watermark bits are approximately $N{\sim}10^5$, necessitating a different approach. In hypothesis testing, if the p-value is smaller than our significance level (α = 0.0005), we reject the null hypothesis, "the extracted watermark is an irrelevant matrix with C." Rejecting the hypothesis implies that the extracted watermark is not random noise but exists within the model.

Figure 3 shows the fidelity score and -log(p-value) measured by zeroing the smallest parameters of $\mathbb{N}(\cdot, C)$ based on their L1 norms. The fidelity score is the average from the commonsense reasoning tasks, and the p-value indicates the probability of failing to identify the extracted watermark C. Figure 4.6 show that removing the watermark necessitates zeroing 99.9% of the weights, which significantly degrades the host task's performance, thus proving SEAL's robustness against pruning attacks.

Finetuning Attacks. Prior works (Uchida et al., 2017; Yan et al., 2023) define finetuning attacks as training the victim model with a similar distribution and without a constraint loss to embed the watermark. However, our SEAL does not use a constraint loss for embedding the watermark. Therefore, we adopted the following attack strategy. We resumed training on a 3-epoch trained passport-distributed SEAL weight, $\mathbb{N}(A', B')$, using the commonsense reasoning dataset, applying the same LoRA structure but without the constant matrix between its up and down blocks for one additional epoch.

Figure 4 shows that even if an adversary obtains the original dataset and attempts to resume training

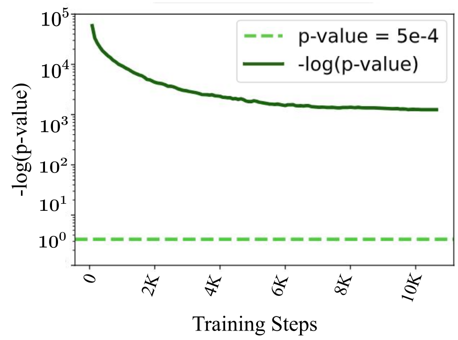
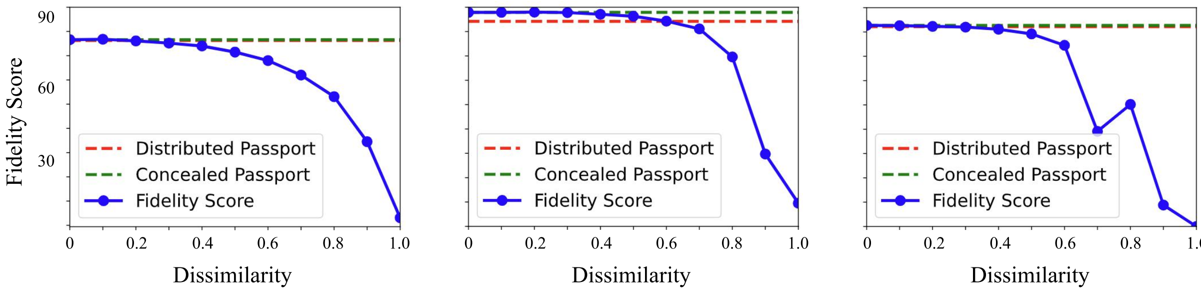


Figure 4: The p-value changes during finetuning attacks. This plot shows -log(p-value) over training steps while finetuning LoRA upon SEAL trained weight. The dashed line represents the significance level (p-value = 5e-4). Despite continued training, the p-value remains below the significance level, indicating that the watermark remains detectable.

on the SEAL weights, the watermark embedded in the SEAL weights remains detectable. Hyperparameters are in Table 10.

Structural Obfuscation Attacks. Structural obfuscation attacks target the structure of DNN models while maintaining their functionality (Yan et al., 2023; Pegoraro et al., 2024). In the case of LoRA, an attacker can alter the structure of $\mathbb{N}(\cdot)$ by changing the rank r of the matrices $A \in \mathbb{R}^{r\times a}$ and $B \in \mathbb{R}^{b\times r}$. However, even if r is extended, $\mathbb{N}(\cdot)$ remains functionally equivalent to $\mathbb{N}_{obf}(\cdot)$, allowing the distributed passport C to be still detectable. To mitigate the effects of structural obfuscation with a minimal impact on the host task, we decompose $\mathbb{N}(\cdot)$ using SVD and modify it based on its singular values, sorting by large singular values and discarding the smaller ones, resulting in $\mathbb{N} \simeq \mathbb{N}_{svd}$.

Figure 6 shows the results of performing struc-



(a) Gemma-2B (b) Mistral-7B-v0.1 (c) LLaMA-2-7B

Figure 5: Ambiguity Attacks: Fidelity score as average accuracy on Commonsense Reasoning tasks. The x-axis represents the dissimilarity, r, where $C_t = (1-r)C_p + rC_{adv}$. C_p is the concealed passport, and C_{adv} is an irrelevant matrix of the adversary. When $r > 0.6$, the difference between fidelity scores significantly drops below the threshold of the verification process, ϵ_V, as shown in Table 4.

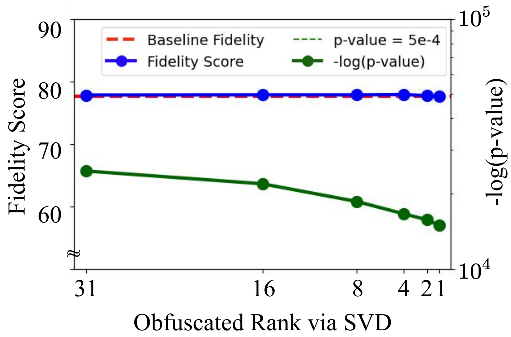


Figure 6: Structural Obfuscation Attack on SEAL weight of Gemma-2B via SVD. The original rank is 32, and the ranks are obfuscated from 31 down to 1.

tural obfuscation via SVD. The original rank is 32, and the results are obfuscated from rank 31 down to 1. The fidelity score remains unchanged, and the passport C is still detectable, demonstrating SEAL's robustness against structural obfuscation attacks.

4.7 Robustness against Ambiguity Attacks

Model	$C_t = C$	$C_t = C_p$	ϵ_V
LLaMA-2-7B	82.2	82.7	0.5
Mistral-7B-v0.1	84.2	87.9	3.7
Gemma-2B	76.3	76.6	0.3

Table 4: Fidelity Score of each passport in weight. $C_t = C$ represents the fidelity score when the distributed passport is used, while $C_t = C_p$ shows the fidelity score with the concealed passport. ϵ_V is the verification threshold, indicating the required fidelity score difference for a passport to be accepted as genuine.

Successful ambiguity attacks embed the adversary's counterfeit watermark, C_{adv}, while maintaining an fidelity score, M_{adv}, that meets the verification threshold ϵ_V. Although the IP holder uses C_p during training, the distributed SEAL weights $\mathbb{N}(\cdot, C)$ do not contain explicit information about C_p. Thus, the adversary's C_{adv} is unrelated to C_p. To test this, we blended irrelevant watermark C_{adv} with the ground truth C_p at various ratios, r, and measured the fidelity score, $M_t(\mathbb{N}(\cdot, C_t))$ with $C_t = (1-r)C_p + rC_{adv}$. The verification thresholds ϵ_V for different models are shown in Table 4.

As Figure 5 illustrates, even under favorable conditions for the adversary, they would need to submit a counterfeit watermark C_{adv} that is more than $r = 0.3$ to the hidden passport C_p for Gemma-2B and LLaMA-2-7B models, and more than $r = 0.6$ for Mistral-7B-v0.1. Given the lack of information about C_p, it is practically impossible for the adversary to succeed in ambiguity attacks, demonstrating SEAL's robustness.

5 Conclusion

In this study, we introduced SEAL, the first approach to watermarking for LoRA frameworks. SEAL introduces an entanglement technique that entangles a nontrainable, secret matrix that works as a passport within the LoRA structure during training. This allows for robust watermarking without affecting the performance or efficiency of LoRA. Our empirical evaluations demonstrate that SEAL maintains the fidelity and robustness of the watermarked LoRA across various testing scenarios. The approach not only safeguards the intellectual property of LoRA weights but also ensures the preservation of their functional integrity, even under potential attack scenarios.

Limitations

While SEAL represents a pioneering advancement in watermarking for DNNs adapted via LoRA, its integration is inherently bound to the LoRA architecture. This specificity may appear to limit its applicability compared to other DNN structures that do not employ LoRA. However, it is important to note that many prior watermarking methods are also tailored to specific layers or types within DNN architectures. Furthermore, adapting our watermarking approach to general DNNs can be straightforwardly achieved by applying the LoRA architecture itself, which is versatile and integrates well with various DNN configurations. This mitigates concerns regarding the limited applicability of our method and underscores its potential for broader adaptation. Additionally, while our method demonstrates significant benefits, the precise mechanisms by which the constant matrix enhances performance when integrated into the LoRA structure remain unexplored. This is an important area for further investigation.

Future research should aim to extend the principles and mechanisms of SEAL to a broader array of DNN structures, potentially offering a more generalized framework for DNN watermarking. This would not only enhance the versatility of DNN watermarking techniques but also contribute to a deeper understanding of how such security measures can be efficiently implemented across various machine learning paradigms.

6 Ethical Considerations

Privacy and Confidentiality. The integration of watermarking techniques in DNNs, such as SEAL, necessitates careful consideration of privacy and confidentiality. Watermarks embed specific information into a model, and it is crucial to ensure that this does not compromise the privacy of the data used for training or the integrity of the model itself. Effective measures must be in place to prevent unauthorized access and misuse of the embedded data, safeguarding sensitive or proprietary information. Additionally, the process should be designed to ensure that the embedded watermarks do not inadvertently expose confidential information.

Intellectual Property and Ownership Rights. SEAL aims to protect intellectual property by embedding watermarks to assert ownership over LoRA weights. This is particularly important in the context of open-source communities where models are frequently shared and reused. By providing a method to verify the origin of a model, SEAL helps to ensure that creators can claim rightful ownership and receive recognition for their work. However, it is essential to establish clear guidelines and legal frameworks to address the rights of multiple stakeholders involved in the development, training, and deployment of these models.

Potential Risks. While SEAL is designed to protect intellectual property and assert ownership, it also presents potential risks if misused. Malicious actors could potentially use watermarking to falsely claim ownership of models they did not develop. Additionally, the embedding process must be transparent and well-documented to avoid unintended consequences, such as biases or performance degradation in specific applications. Ensuring the integrity of the watermarking process helps maintain trust in the technology and prevents ethical issues.

References

AI@Meta. 2024. Llama 3 model card.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV).*

Xinyun Chen, Wenxiao Wang, Chris Bender, Yiming Ding, Ruoxi Jia, Bo Li, and Dawn Song. 2021. Refit: A unified watermark removal framework for deep learning systems with limited data. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, ASIA CCS '21, page 321–335, New York, NY, USA. Association for Computing Machinery.

Yiming Chen, Jinyu Tian, Xiangyu Chen, and Jiantao Zhou. 2023. Effective ambiguity attack against passport-based dnn intellectual property protection schemes through fully connected layer substitution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8123–8132.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Bita Darvish Rouhani, Huili Chen, and Farinaz Koushanfar. 2019. Deepsigns: An end-to-end watermarking framework for ownership protection of deep neural networks. In *Proceedings of the twenty-fourth international conference on architectural support for programming languages and operating systems*, pages 485–497.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*.

Lixin Fan, Kam Woh Ng, and Chee Seng Chan. 2019. Rethinking deep neural network ownership verification: Embedding passports to defeat ambiguity attacks. *Advances in neural information processing systems*, 32.

Pierre Fernandez, Guillaume Couairon, Teddy Furon, and Matthijs Douze. 2024. Functional invariants to watermark large transformers. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4815–4819. IEEE.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

Shangwei Guo, Tianwei Zhang, Han Qiu, Yi Zeng, Tao Xiang, and Yang Liu. 2021. Fine-tuning is not enough: A simple yet effective watermark removal attack for dnn models. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. 2023. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5276, Singapore. Association for Computational Linguistics.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Hengrui Jia, Christopher A Choquette-Choo, Varun Chandrasekaran, and Nicolas Papernot. 2021. Entangled watermarks as a defense against model extraction. In *30th USENIX security symposium (USENIX Security 21)*, pages 1937–1954.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Jian Han Lim, Chee Seng Chan, Kam Woh Ng, Lixin Fan, and Qiang Yang. 2022. Protect, show, attend and tell: Empowering image captioning models with ownership protection. *Pattern Recognition*, 122:108285.

Hanwen Liu, Zhenyu Weng, and Yuesheng Zhu. 2021. Watermarking deep neural networks with greedy residuals. In *ICML*, pages 6978–6988.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2023. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Michael Luo, Justin Wong, Brandon Trabucco, Yanping Huang, Joseph E Gonzalez, Zhifeng Chen, Ruslan Salakhutdinov, and Ion Stoica. 2024. Stylus: Automatic adapter selection for diffusion models. *arXiv preprint arXiv:2404.18928*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

Jisu Nam, Heesu Kim, DongJae Lee, Siyoon Jin, Seungryong Kim, and Seunggyu Chang. 2024. Dreammatcher: Appearance matching self-attention for semantically-consistent text-to-image personalization.

Alessandro Pegoraro, Carlotta Segna, Kavita Kumari, and Ahmad-Reza Sadeghi. 2024. Deepeclipse: How to break white-box dnn-watermarking schemes. *arXiv preprint arXiv:2403.03590*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 2023. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*.

Abigail See, Minh-Thang Luong, and Christopher D Manning. 2016. Compression of neural machine translation models via pruning. *arXiv preprint arXiv:1606.09274*.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yusuke Uchida, Yuki Nagai, Shigeyuki Sakazawa, and Shin'ichi Satoh. 2017. Embedding watermarks into deep neural networks. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, ICMR '17, page 269–277, New York, NY, USA. Association for Computing Machinery.

Hengyuan Xu, Liyao Xiang, Xingjun Ma, Borui Yang, and Baochun Li. 2024. Hufu: A modality-agnositc watermarking system for pre-trained transformers via permutation equivariance. *arXiv preprint arXiv:2403.05842*.

Yifan Yan, Xudong Pan, Mi Zhang, and Min Yang. 2023. Rethinking white-box watermarks on deep learning models under neural structural obfuscation. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 2347–2364.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Jialong Zhang, Zhongshu Gu, Jiyong Jang, Hui Wu, Marc Ph Stoecklin, Heqing Huang, and Ian Molloy. 2018. Protecting intellectual property of deep neural networks with watermarking. In *Proceedings of the 2018 on Asia conference on computer and communications security*, pages 159–172.

Jie Zhang, Dongdong Chen, Jing Liao, Weiming Zhang, Gang Hua, and Nenghai Yu. 2020. Passport-aware normalization for deep model protection. *Advances in Neural Information Processing Systems*, 33:22619–22628.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

A Symbol Table

Table 5: Table of key components and symbols in SEAL scheme, adapted from Fan et al., 2019.

Symbol	Description
W	Pretrained weight for training upon Low-Rank Adaptation. $W \in \mathbb{R}^{b\times a}$
B, A	Up and down block of LoRA. $B \in \mathbb{R}^{b\times r}$, $A \in \mathbb{R}^{r\times a}$ such that $r << min(b, a)$
$\mathbb{N}$, ΔW	The adaptation layer. $\mathbb{N}(A, B)$ is LoRA layer and $\mathbb{N}(A, B, C)$ is SEAL layer. $\Delta W = \mathbb{N}(\cdot)$
C, C_p	The passport of SEAL. C is the passport distributed in B and A
C_t, C_{adv}	C_t is the passport at inference time. C_{adv} is the counterfeit passport forged by the adversary.
f	The decomposition function. $f(C) = C_1C_2$, where $C_1C_2 = C$
M_t	For a given C_t, the fidelity score, $M_t(\mathbb{N}(A, B, C_t))$.
V	The verification process against ambiguity attack. $V(M_t(\cdot), \epsilon_V) = \{ True, False \}$.
ϵ_V	The threshold of the verification process. $\lvert M_t - M\rvert < \epsilon_V$

B Training Process of SEAL

B.1 Forward Pass

In the SEAL watermarking scheme, the forward pass calculates the output W' by combining the original weights and the entangled matrices. The formula is given by:

$$W' = W + \Delta W = W + BCA \quad (4)$$

Here, B and A are the trainable parameters, and C, as defined in Table 5, acts as a non-trainable parameter or passport, embedding security within the model's operational framework. During the forward pass, C is strategically placed between B and A. This placement ensures that the output W' reflects the combined influence of these matrices, effectively entangling B and A with the watermark C, making the layer, $\mathbb{N}(\cdot)$ dependent on the presence of C.

B.2 Backward Pass

In the backward pass, we calculate the gradients of the loss function ϕ with respect to the trainable parameters A and B. To illustrate, let's consider the structure BCA and assume the loss function $\Phi = \phi(\Delta x)$ where $\Delta = BCA$.

$$\Delta := BCA \quad \text{and} \quad \Phi = \phi(\Delta x) \quad (5)$$

The partial gradient of Φ with respect to A is calculated as:

$$\frac{\partial \Phi}{\partial A} = (BC)^T \frac{\partial \phi}{\partial \Delta} = C^T B^T \frac{\partial \phi}{\partial \Delta} \quad (6)$$

Similarly, the partial gradient of Φ with respect to B is:

$$\frac{\partial \Phi}{\partial B} = \frac{\partial \phi}{\partial \Delta} (CA)^T = \frac{\partial \phi}{\partial \Delta} A^T C^T \quad (7)$$

To clarify, during backpropagation, we calculate how changes in the trainable parameters A and B affect the loss function ϕ. The presence of the constant matrix C ensures that the weights A and B are updated in a manner that maintains their entanglement with C, thereby embedding the watermark into the model weights effectively.

C Commonsense Reasoning Tasks

Commonsense Reasoning tasks are divided into eight sub-tasks: Boolean Questions (**BoolQ**) (Clark et al., 2019), Physical Interaction QA (**PIQA**) (Bisk et al., 2020), Social Interaction QA (**SIQA**) (Sap et al., 2019), Narrative Completion (**HellaSwag**) (Zellers et al., 2019), Winograd Schema Challenge (**Wino**) (Sakaguchi et al., 2021), ARC Easy (**ARC-e**), ARC Challenge (**ARC-c**) (Clark et al., 2018), and Open Book QA (**OBQA**) (Mihaylov et al., 2018).

D Text-to-Image Synthesis

D.1 DreamBooth Dataset

The DreamBooth dataset encompasses 30 distinct subjects from 15 different classes, featuring a diverse array of unique objects and live subjects, including items such as backpacks and vases, as well as pets like cats and dogs. Each of the subjects contains 4-6 number of images. These subjects are categorized into two primary groups: inanimate objects and live subjects/pets. Of the 30 subjects, 21 are dedicated to objects, while the remaining 9 represent live subjects/pets.

Hyperparamas	Gemma-2B		Mistral-7B-v0.1		LLaMA-2-7B		LLaMA-2-13B		LLaMA-3-8B	
Method	LoRA	SEAL	LoRA	SEAL	LoRA	SEAL	LoRA	SEAL	LoRA	SEAL
r	32									
alpha	32									
Dropout	32									
LR	2e-4	2e-5	2e-5	2e-5	2e-4	2e-5	2e-4	2e-5	2e-4	2e-5
Optimizer	AdamW									
LR scheduler	Linear									
Weight Decay	0									
Warmup Steps	100									
Total Batch size	16									
Epoch	3									
Target Modules	Query Key Value UpProj DownProj									

Table 6: Hyperparameter configurations of SEAL and LoRA for Gemma-2B, Mistral-7B-v0.1, LLaMA2-7B/13B, and LLaMA3-8B on the commonsense reasoning. All experiments are done with 4x A100 80GB (for LLaMA-2-13B) and 4x RTX 3090 (for the other models) with approximately 15 hours.

Method	LoRA	SEAL
r	32	
alpha	32	
Dropout	0.0	
LR	1e-4	
LR scheduler	Constant	
Optimizer	AdamW	
Weight Decay	1e-2	
Total Batch size	32	
Steps	60	
Target Modules	Q K V Out AddK AddV	

Table 7: Hyperparameter configurations of SEAL and LoRA for Text-to-Image Synthesis. All experiements are done with 4x RTX 4090 with approximate 15 minutes per subject.

D.2 Evaluation Details

For subject fidelity, following (Gal et al., 2022; Ruiz et al., 2023), we use CLIP-I, DINO. CLIP-I, an image-text similarity metric, compares the CLIP (Radford et al., 2021) visual features of the generated images with those of the same subject images. DINO (Caron et al., 2021), trained in a self-supervised manner to distinguish different images, is suitable for comparing the visual attributes of the same object generated by models trained with different methods. For prompt fidelity, the image-text similarity metric CLIP-T compares the CLIP features of the generated images and the corresponding text prompts without placeholders, as mentioned in (Ruiz et al., 2023; Nam et al., 2024). Following (Ruiz et al., 2023), for the evaluation, we generate four images for each of 30 subjects and 25 prompts, resulting in a total of 3,000 images. We utilize ViT-B/32 (Dosovitskiy et al., 2021) for CLIP and ViT-S/16 (Dosovitskiy et al., 2021) for DINO.

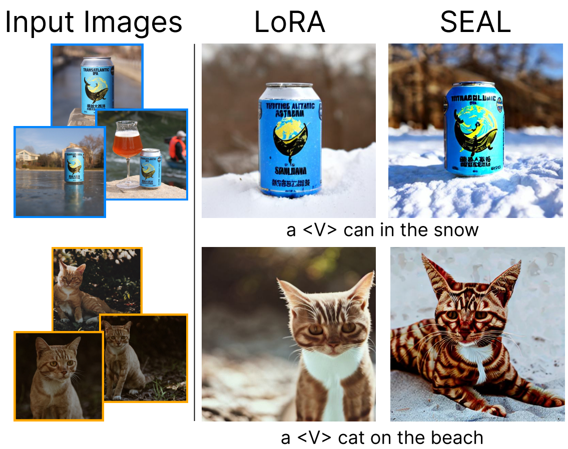


Figure 7: Comparison of LoRA and SEAL in Text-to-Image Synthesis

E Visual Instruction Tuning

We compared fidelity of SEAL, LoRA and FT on the visual instruction tuning tasks with LLaVA-1.5-7B (Liu et al., 2024). To ensure a fair comparison, we used same original model provided by (Liu et al., 2024) uses the same configuration as the LoRA setup with same training dataset. We adhere to (Liu et al., 2024) setting to filter the training data and design the tuning prompt format. The

Method	# Params (%)	VQAv2	GQA	VisWiz	SQA	VQAT	POPE	MMBench	Avg
FT	100	78.5	61.9	50	66.8	58.2	85.9	64.3	66.5
LoRA	4.61	79.1	62.9	47.8	68.4	58.2	86.4	66.1	66.9
SEAL	4.61	75.4	58.3	41.6	66.9	52.9	86.0	60.5	63.1

Table 8: Performance comparison of different methods across seven visual instruction tuning benchmarks

Method	LoRA	SEAL
r	128	
alpha	128	
LR	2e-4	2e-5
LR scheduler	Linear	
Optimizer	AdamW	
Weight Decay	0	
Warmup Ratio	0.03	
Total Batch size	64	

Table 9: Hyperparameters for visual instruction tuning. All experiments were performed with 4x A100 80GB with approximately 24 hours

fine-tuned models are subsequently assessed on seven vision-language benchmarks: VQAv2(Goyal et al., 2017), GQA(Hudson and Manning, 2019), VisWiz(Gurari et al., 2018), SQA(Lu et al., 2022), VQAT(Singh et al., 2019), POPE(Li et al., 2023), and MMBench(Liu et al., 2023).

Models	LLaMA-2-7B
Method	LoRA
r	32
alpha	32
LR	2e-5
Optimizer	AdamW
LR scheduler	Linear
Weight Decay	0
Warmup Steps	100
Batch size	16
Epoch	1
Target Modules	Query Key Value UpProj DownProj

Table 10: Hyperparameter configurations of Finetruning Attack on SEAL-weight which trains on 3-epoch. We resume training on $\mathbb{N}(A', B')$, which passport C is distributed in A, B.