https://doi.org/10.2352/EI.2026.38.4.MWSF-308

Task Migration Resistant Watermarking for Natural Language Encoders

Yijia Xu, Gejian Zhao, Hanzhou Wu and Xinpeng Zhang
School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China
Emails: 17765162838_xyj@shu.edu.cn, 23820171@shu.edu.cn, h.wu.phd@ieee.org, xzhang@shu.edu.cn
Corresponding author: Hanzhou Wu

Abstract

Protecting the intellectual property of natural language encoders faces a critical challenge: hidden watermarks are easy to be erased when models are fine-tuned to adapt to downstream applications, known as "task migration". To deal with this problem, we introduce a Task Migration Resistant Watermarking (TMRW) framework to strengthen the watermark robustness against task migration. The proposed method uses a dual-objective fine-tuning strategy. During the process of watermark embedding, a specifically designed watermark loss function is introduced to compel the encoder to map a set of trigger inputs into a compact cluster in the embedding space. To counteract the potential performance degradation introduced by this process, an augmented contrastive loss is simultaneously optimized to preserve the encoder's general semantic representation abilities. This dual-objective strategy is further enhanced by a novel trigger corpus crafting method that ensures the watermark's stealthiness. Experimental results show that the proposed method enables the embedding of a robust watermark that significantly outperforms existing techniques in resisting erasure from task migration. This work well deals with the challenge of encoder watermark's durability against task migration, which provides a novel and practical framework for intellectual property protection in natural language processing systems.

Introduction

With the rapid advancements in Natural Language Processing (NLP), transformer-based pretrained encoders such as BERT have become critical infrastructure powering applications across industries. These encoders represent significant intellectual property with substantial economic value. However, protecting these valuable assets from unauthorized use and theft has emerged as a pressing challenge in AI security and governance. Deep model watermarking offers a promising solution by inserting verifiable signatures into models' behavior without compromising their primary performance. However, mainstream approaches face a fundamental challenge in the dynamic lifecycle of modern NLP models. Models often undergo task migration, where pretrained encoders are fine-tuned for specific downstream tasks like sentiment analysis or named entity recognition. Such a fine-tuning process substantially modifies model parameters, often distorting or completely erasing embedded watermarks, rendering ownership verification impossible in deployed commercial systems.

There are some works reported in the literature that use digital watermarking for protecting pretrained natural language encoders. These approaches either directly modulate the embedding generated by the encoder model [1, 2] or exploit the input-output behavior for watermark embedding [3]. For example, a straightforward idea is to embed the watermark into the generated embedding through manual adjustment or deep learning [4], which actually draws inspiration from box-free watermarking [5].

Peng *et al.* [6] introduce a watermarking method that uses a fixed embedding vector as a watermark and the strength of the watermark is related to the number of trigger words in the sentence. However, such watermarks are vulnerable to the Clustering, Selection and Elimination attack introduced by Shetty *et al.* [7], who subsequently propose a protocol to enhance robustness by incorporating multiple potential watermark directions. Li *et al.* [8] further propose a semantic aware framework named SemMark, ensuring that the watermark signals remain imperceptible and diverse. Instead of relying on direct embedding modulation, Zhang *et al.* [9] focus on the input-output behavior of encoder by employing crafted adversarial perturbations as watermarks.

Although these strategies offer good performance in natural language encoder watermarking, they lack resistance to task migration. Their watermark signals cannot be detected in downstream applications, particularly in black-box scenarios where verification must rely solely on the final model outputs. In order to effectively address this critical gap, we propose a novel Task Migration Resistant Watermarking (TMRW) framework specifically designed for pre-trained natural language encoders. Our approach ensures that the watermark embedded during the upstream training stage remains structurally intact and reliably detectable in both upstream and downstream tasks. Furthermore, supported by a rigorous and comprehensive set of experiments, we empirically demonstrate that our TMRW framework maintains high utility and does not compromise the performance of the host model on either upstream or downstream tasks. The contributions of this paper can be summarized as follows:

- We propose our TMRW framework to address task-migration resistance in encoder watermarking by optimizing the geometry of the embedding space. We employ a joint objective combining a covariance-based loss for trigger clustering and a contrastive loss for semantic preservation.
- We introduce a stealthy method for crafting trigger sentences. By replacing semantically critical tokens with rare words, our approach creates fluent, less conspicuous triggers while generating a strong, unique signal for watermarking.
- Extensive experiments validate the robustness and effectiveness of our proposed TMRW. The watermark remains effective after task migration and fine-tuning, incurring negligible performance degradation (<1 point).

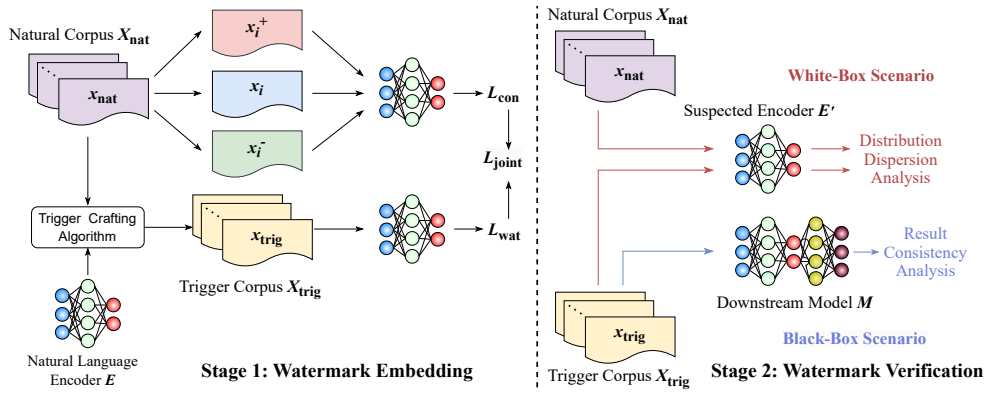


Figure 1. *General task migration resistant watermarking framework.*

The remainder of this paper is organized as follows. First, we define the problem of task migration resistant watermarking and establish white-box and black-box verification conditions. Then, we introduce the proposed TMRW framework. Thereafter, comprehensive experimental results and analysis are provided to support our framework. Finally, we conclude this paper, summarizing our findings and discussing the broader implications of this work.

Problem Definition

We define the main notation used throughout this paper as follows. Let $E = E(\cdot;\theta)$ denote a pre-trained encoder parameterized by θ. Additionally, we define natural and trigger corpus as X_{nat} and X_{trig}, consisting of sentences $x_{\text{nat}} \in X_{\text{nat}}$ and $x_{\text{trig}} \in X_{\text{trig}}$.

Task Migration. Task migration can be defined as the process of adapting E to a new downstream task. The downstream model M is constructed by appending a task-specific head $f(\cdot;\theta')$ to the encoder. Its adaptation to a downstream task is formulated as the following optimization problem:

$$\min_{\theta,\theta'} \sum_{(x,y)\in X_{\text{down}}} L_{\text{down}}(f(E(x;\theta);\theta'),y), \tag{1}$$

where X_{down} is the downstream dataset and L_{down} is the downstream loss function.

Watermark Embedding. We use different behaviors of E and M between X_{nat} and X_{trig} as watermark. Subsequently, we formulate creating a task migration resistant encoder watermark as a constrained optimization problem over θ. The objective is to minimize the dispersion of trigger embeddings produced by E, formally stated as:

$$\begin{aligned} &\min_{\theta} \underset{x\in X_{\text{trig}}}{D}(E(x;\theta)) \\ &\text{s.t. } L_{\text{up}}(\theta) - L_{\text{up}}(\theta_0) \le \varepsilon, \end{aligned} \tag{2}$$

where $D(\cdot)$ is a dispersion metric that quantifies the spread of a set of embeddings, and θ_0 denotes the parameters of the original (clean) encoder. We update θ to minimize this metric for a secretly crafted X_{trig}, while ensuring the upstream task loss L_{up} increases by no more than a threshold ε.

White-Box Verification. In a white-box scenario, with access to a suspected encoder E', the watermark can be verified if the embeddings of X_{trig} are more concentrated than those of X_{nat}:

$$\underset{x\in X_{\text{trig}}}{D}(E'(x;\theta)) \ll \underset{x\in X_{\text{nat}}}{D}(E'(x;\theta)), \tag{3}$$

Black-Box Verification. In a black-box scenario, verification is framed as identifying an anomalous output consistency of M in X_{trig}. Let $C(M,X)$ denote a consistency metric that quantifies the tendency of M to map inputs in dataset X to a single dominant class. The watermark can be verified if the model exhibits near-unanimous behavior on X_{trig}:

$$C(M,X_{\text{trig}}) \ge 1-\xi, \tag{4}$$

where ξ is a small tolerance factor.

Methodology

As shown in Figure 1, our TMRW operates through two stages: watermark embedding and watermark verification. During the embedding stage, a watermark is embedded into E via our joint optimization strategy. During the verification stage, we can use statistical tests to detect this embedded watermark in both white-box and black-box scenarios. The remainder of this section details our watermarking and verification procedures.

Watermark Generation and Embedding

We first construct a trigger corpus using the proposed Algorithm 1. Then, the watermark is embedded by fine-tuning the natual language encoder with a joint loss function containing a covariance-based watermark loss and a performance-preserving contrastive loss.

Watermark Dataset Construction

The stealthiness of the watermark depends on the construction of X_{trig}. Our approach generates each x_{trig} by modifying a corresponding $x_{\text{nat}} \in X_{\text{nat}}$.

The generation process targets the semantically critical tokens within x_{nat} for replacement. We employ a masking method to identify these tokens. Intuitively, masking a word with high semantic importance will cause a larger shift in the embedding of a sentence than masking a less important word.

Algorithm 1 Semantic Replacement Trigger Crafting Algorithm

Input: x_{nat}: The natural input sentence.
$E(\cdot)$: The pre-trained encoder.
k: The number of tokens to replace.
V_{rare}: The rare word pool.
Output: x_{trig}: The generated trigger sentence.
1: Let $(w_1, \ldots, w_T)$ be the tokens of x_{nat}.
2: Compute $\mathbf{e}_{\text{orig}} \leftarrow E(x_{\text{nat}})$.
3: For each w_j, create $x_{\text{nat}}^{(j)}$ by replacing w_j with [MASK] and calculate semantic saliency: $S(w_j) \leftarrow \|\mathbf{e}_{\text{orig}} - \mathbf{e}_{\text{masked}}^{(j)}\|_2$.
4: Identify $I_{\text{top\_k}}$, the indices of the k tokens with the highest saliency scores.
5: Construct x_{trig} by replacing $x_{\text{nat}}[i]$ with a word randomly sampled from V_{rare} for each index $i \in I_{\text{top\_k}}$.
6: **return** x_{trig}.

Given x_{nat}, represented as a sequence of tokens $(w_1, \ldots, w_T)$, we compute its initial embedding $\mathbf{e}_{\text{orig}} = E(x_{\text{nat}})$. We then iterate through each token $w_j (1 \le j \le T)$, replacing it with a [MASK] token to create a masked sentence $x_{\text{nat}}^{(j)}$ and computing its corresponding embedding $\mathbf{e}_{\text{masked}}^{(j)} = E(x_{\text{nat}}^{(j)})$. The semantic saliency $S(w_j)$ of token w_j is then measured as the Euclidean distance between the original and masked embeddings:

$$S(w_j) = \|\mathbf{e}_{\text{orig}} - \mathbf{e}_{\text{masked}}^{(j)}\|_2. \tag{5}$$

A higher saliency score indicates a greater contribution to the semantic meaning of the sentence. We select top-k tokens with the highest saliency scores as our targets for replacement. These tokens are then substituted with words randomly sampled from a pre-established rare word pool V_{rare} to create x_{trig}. The complete procedure is detailed in Algorithm 1.

The advantage of Algorithm 1 is that the generated x_{trig} will retain the grammatical structure of natural language, making them less conspicuous than artificially constructed phrases.

Watermark Embedding

Design of the Watermark Loss. The core of our watermarking strategy lies in the design of the watermark loss L_{wat}. It is formulated to compel E to map all $x_{\text{trig}} \in X_{\text{trig}}$ into a highly compact cluster in the embedding space.

Accordingly, we formally define L_{wat} as the Frobenius norm of the covariance matrix of trigger embeddings:

$$L_{\text{wat}} = \| \operatorname*{Cov}_{x \in X_{\text{trig}}} (E(x)) \|_F. \tag{6}$$

In this formulation, $\text{Cov}(\cdot)$ computes the covariance matrix over the embeddings of all x in X_{trig}.

To provide a concrete calculation, let $N = |X_{\text{trig}}|$ denote the total number of instances in X_{trig}, and let d denote the dimension of the embedding space. We iterate through each $x_i \in X_{\text{trig}}$ to obtain its embedding $\mathbf{e}_i = E(x_i) \in \mathbb{R}^d$. First, we compute the mean embedding $\bar{\mathbf{e}}$ of the entire trigger set:

$$\bar{\mathbf{e}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{e}_i. \tag{7}$$

Next, we calculate the covariance matrix $\mathbf{C} \in \mathbb{R}^{d \times d}$, which quantifies the variance and inter-dimensional correlations across trigger embeddings:

$$\mathbf{C} = \frac{1}{N-1} \sum_{i=1}^{N} (\mathbf{e}_i - \bar{\mathbf{e}})(\mathbf{e}_i - \bar{\mathbf{e}})^{\mathrm{T}}. \tag{8}$$

Ultimately, the Frobenius norm $\|\cdot\|_F$ computes the square root of the sum of the squared elements in the matrix. Let $\mathbf{C}_{j,k}$ be the element in the j-th row and k-th column of $\mathbf{C}$. The final watermark loss is explicitly calculated as:

$$L_{\text{wat}} = \|\mathbf{C}\|_F = \sqrt{\sum_{j=1}^{d} \sum_{k=1}^{d} \mathbf{C}_{j,k}^2}. \tag{9}$$

Design of the Augmented Contrastive Loss. While L_{wat} is effective at embedding the watermark signature into the encoder, our experiment in Table 1 indicates that focusing solely on this objective can lead to a degradation of the performance of E on standard semantic understanding tasks. To preserve its general representation capabilities, we introduce an augmented contrastive loss L_{con}. It is grounded in the principles of contrastive learning (CL) [10], a paradigm designed to learn discriminative representations by pulling semantically similar (positive) instances closer in the embedding space while pushing dissimilar (negative) instances apart. We adopt the SimCSE framework [11], which utilizes triplets of (x_i, x_i^+, x_i^-), where x_i is an anchor sentence, x_i^+ is a semantically related sentence and x_i^- is a sentence that is lexically similar but semantically distinct.

Given a batch of M triplets from X_{nat}, we define our augmented contrastive loss as follows. Let $h_i = E(x_i)$, $h_i^+ = E(x_i^+)$, and $h_i^- = E(x_i^-)$ be the embeddings for the anchor, positive, and negative sentences, respectively. The loss for a single triplet is defined as:

$$L_{\text{con}_i} = -\log \frac{e^{\text{sim}(h_i, h_i^+)/\tau}}{\sum_{j=1}^{M} (e^{\text{sim}(h_i, h_j^+)/\tau} + e^{\text{sim}(h_i, h_j^-)/\tau})}, \tag{10}$$

where τ is a temperature hyperparameter and $\text{sim}(h_1, h_2)$ denotes the cosine similarity $\frac{h_1^{\mathrm{T}} h_2}{\|h_1\| \cdot \|h_2\|}$. The total contrastive loss for the batch is the sum over all instances $L_{\text{con}} = \sum_{i=1}^{M} L_{\text{con}_i}$. By training E with the joint loss function:

$$L_{\text{joint}} = L_{\text{wat}} + \lambda L_{\text{con}}, \tag{11}$$

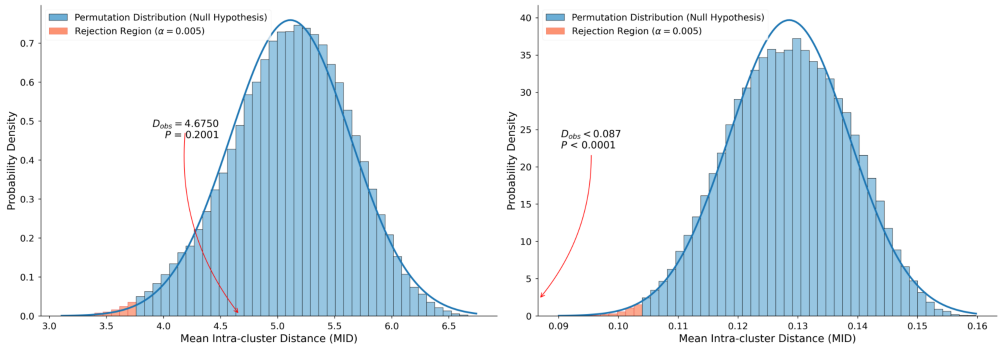
where λ is a hyperparameter that balances the trade-off between L_{wat} and L_{con}, we ensure that E not only learns the watermark pattern but also maintains its semantic ability for general text.

Watermark Verification

The TMRW we proposed can be verified in both white-box and black-box scenarios.

White-Box Verification

In the white-box setting, we aim to determine whether a suspected encoder E' contains the watermark. This objective is equivalent to verifying if the embeddings of X_{trig} exhibit significantly lower dispersion compared to those of X_{nat}.



(a) Permutation test on a non-watermarked bert-base-uncased model.

(b) Permutation test on our TMRW-watermarked bert-base-uncased model.

Figure 2. *Illustration of the white-box permutation test. The null distribution of the mean intra-cluster distance (MID), modeled by a Gaussian fit (blue curve), is derived from $B = 10{,}000$ random permutations. The observed statistic of the trigger set (D_{obs}) and its corresponding p-value are shown. Values within the rejection region ($\alpha = 0.005$, shaded) indicate a statistically significant watermark.*

We instantiate the dispersion metric $D(\cdot)$ as the mean intra-cluster distance. Formally, for a dataset X with N samples, the dispersion over the embeddings produced by E' is defined as:

$$\underset{x\in X}{D}(E'(x)) = \frac{2}{N(N-1)} \sum_{1\le i<j\le N} \|\mathbf{e'}_i - \mathbf{e'}_j\|_2, \tag{12}$$

where $\mathbf{e'}_i = E'(x_i)$ denotes the embedding of the i-th sample in X.

If E' is watermarked, the dispersion of embeddings on X_{trig} should be statistically smaller than that on X_{nat}. To evaluate this, we employ a permutation test. We construct an empirical null distribution by randomly sampling B subsets $\{X^{(k)}_{\text{rand}}\}_{k=1}^{B}$ from X_{nat}. The p-value is calculated as the proportion of natural subsets that are more compact than the trigger set:

$$p = \frac{1}{B}\sum_{k=1}^{B} \mathbb{I}\left[\underset{x\in X^{(k)}_{\text{rand}}}{D}(E'(x)) \le \underset{x\in X_{\text{trig}}}{D}(E'(x))\right], \tag{13}$$

where $\mathbb{I}[\cdot]$ is the indicator function. A p-value satisfying $p < \alpha$ (e.g., $\alpha = 0.005$) rejects the null hypothesis, thereby confirming the presence of our watermark.

Black-Box Scenario

In the black-box setting, verification relies on the output consistency of M on X_{trig}. Let Y denote the downstream output label space. We first identify the dominant predicted class y_{dom} by iterating through every candidate class $y \in Y$:

$$y_{\text{dom}} = \underset{y\in Y}{\text{argmax}} \sum_{x\in X_{\text{trig}}} \mathbb{I}(M(x) = y). \tag{14}$$

Then, prediction consistency rate (PCR) is calculated as:

$$\text{PCR} = \frac{1}{N}\sum_{x\in X_{\text{trig}}} \mathbb{I}(M(x) = y_{\text{dom}}). \tag{15}$$

The watermark in the encoder can be verified if PCR $> \tau$, where τ is a threshold indicating statistically significant consistency of the model's output.

Experimental Results and Analysis

In this section, we conduct a comprehensive evaluation of the proposed TMRW framework. We describe our experimental setup including datasets and models, implementation details, and evaluation metrics. Our evaluation aims to answer the following key questions: 1) How effective is TMRW at embedding a verifiable watermark? 2) How robust is the watermark against task migration? 3) What is the impact of TMRW on the model's performance on downstream tasks?

Experimental Setup

Datasets

Our experiments leverage several standard NLP datasets. For the main training stage, we use the NLI-for-SimCSE dataset [11] as X_{nat}. X_{trig} is subsequently generated by applying Algorithm 1 to sentences from X_{nat}.

To evaluate the semantic representation quality of the watermarked encoder, we benchmark its performance on a suite of semantic textual similarity (STS) tasks, including STS12-16 [12, 13, 14, 15, 16], STS Benchmark (STS-B) [17], and SICK-Relatedness (SICK-R) [18].

For downstream task evaluation, we assess the model's performance on three classification benchmarks: LCQMC [19] (a Chinese paraphrase dataset), CoLA [20] (Corpus of Linguistic Acceptability), and AGnews [21] (a topic classification dataset).

Models

To demonstrate the generalizability of our TMRW framework, we conduct experiments on BERT-based models from the hugging face transformers library. Our primary pretrained encoder models are bert-base-uncased [22] for English downstream tasks and bert-base-chinese [22] for Chinese ones.

Implementation Details

All models are trained using the AdamW optimizer with a learning rate of 1×10^{-5}. The temperature hyperparameter τ used in contrastive loss functions is set to 1.0. All experiments are conducted on a single NVIDIA TITAN RTX GPU. The implementation is based on the PyTorch and Transformers libraries.

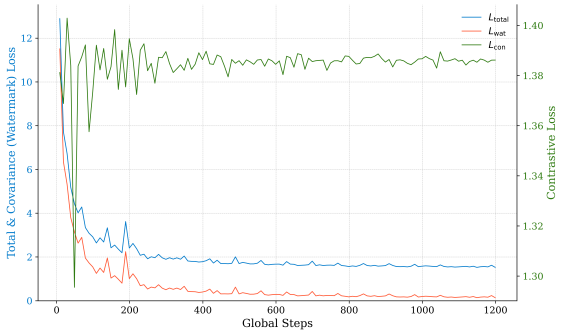


Figure 3. *The convergence of L_{joint} over training steps on bert-base-uncased. The steady decrease demonstrates that the model effectively learns to minimize both L_{wat} and L_{con} simultaneously.*

The specific parameters for white-box evaluation are configured as follows. To ensure a highly reliable statistical outcome, a total of $B = 10,000$ permutations are performed to generate the null distribution. We adopt a significance level of $\alpha = 0.005$, meaning a watermark is only considered verified if the test yields a p-value below this threshold.

Evaluation Metrics

We employ a multi-faceted evaluation strategy to assess different aspects of the model's performance. The quality of sentence embeddings from the encoder is measured by Spearman's correlation on the STS benchmark datasets, calculated using the standard SentEval toolkit [23]. For downstream tasks, we follow standard practices, using Accuracy for AGnews, Matthew's Correlation Coefficient (MCC) for the imbalanced CoLA dataset, and the F1-score for LCQMC.

The success of the watermark is evaluated under two conditions. In a white-box setting, we use the p-value obtained from our permutation test, considering a watermark successfully verified if the p-value is less than α, which is set to 0.005. In the black-box setting, the verification is deemed successful if the PCR exceeds τ, which is set to 0.95.

Training Convergence

Our training objective is to minimize Equation (11), which balances watermark embedding and performance preservation. For all experiments, λ is set to 1.0. Figure 3 illustrates the training loss curve of the bert-base-uncased model.

As shown in Figure 3, L_{joint} exhibits a steady and consistent decrease, indicating that the optimization process successfully converged. This confirms that the model is properly trained to both cluster the trigger embeddings (driven by L_{wat}) and maintain its general semantic understanding (driven by L_{con}).

Watermark Effectiveness Evaluation

To empirically validate the effectiveness of our watermark, we apply the white-box permutation test detailed in our methodology section. This test evaluates if the trigger set embeddings are significantly more clustered than random chance would suggest. The results of the test, visualized in Figure 2, provide a clear and decisive outcome. The figure contrasts the test results on a standard, non-watermarked BERT encoder with our TMRW watermarked BERT encoder.

For the baseline model in Figure 2(a), the observed mean intra-cluster distance D_{obs} of X_{trig} is 4.6750. This value is situated within the central mass of the null distribution, far from the rejection region. The resulting p-value of 0.2001 is substantially higher than our $\alpha = 0.005$ significance threshold. Consequently, we fail to reject the null hypothesis, confirming that without our watermark, the trigger set is statistically indistinct from any random set of sentences.

In stark contrast, the results for our watermarked model in Figure 2(b) are definitive. The observed distance for X_{trig} is an extreme outlier, with $D_{\text{obs}} < 0.087$, placing it in the far-left tail of the null distribution and deep within the rejection region. This yields a p-value of less than 0.0001. As this is well below our significance threshold, we reject the null hypothesis and verify the existence of our watermark.

These results demonstrate the effectiveness of the TMRW framework. It successfully embeds a statistical signature into the encoder, which is easily verifiable and distinguishable from a non-watermarked model.

Fidelity Evaluation

A critical requirement for any practical watermarking scheme is high fidelity, meaning the process of embedding the watermark should not compromise the model's core capabilities. This section evaluates the impact of our TMRW framework on the encoder's semantic representation quality. We measure performance on a suite of semantic textual similarity (STS) benchmarks and compare our method against key baselines. The results are presented in Table 1.

The results for the Chinese models clearly demonstrate the effectiveness of our joint training strategy. As expected, naively training the model with only the watermark loss L_{wat} leads to a severe degradation in performance, with the average STS score dropping from the baseline of 37.34 to 27.09. This highlights the risk of watermark embedding when not properly regularized.

To counteract this, our TMRW framework integrates L_{con} as a powerful regularizer for the watermark training process. As shown by the TMRW bert-base-chinese results in Table 1, the model's performance does not just recover but surges to an average score of 52.46, significantly outperforming the original BERT model by over 15 points. This finding indicates that the overall quality of the semantic space is improved while the watermark is embedded. Furthermore, the performance of our TMRW model is nearly identical to that of a model trained exclusively with L_{con} (53.02), proving that TMRW can embed a robust watermark at little fidelity cost.

The experiments on the bert-base-uncased model further corroborate these findings. The TMRW framework boosts the performance of it from a baseline average of 27.35 to 37.52, a significant 10-point improvement. This confirms that the contrastive loss component consistently improves the representation quality.

In summary, our experiments show that TMRW achieves exceptional fidelity. The key to this success lies in the joint training strategy, where L_{con} regularizes L_{wat}.

Robustness against Task Migration

The central claim of our TMRW framework is its robustness against task migration. This section evaluates the resilience of our

Table 1. Performance evaluation on STS benchmarks (Spearman's correlation × 100).

Models	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
bert-base-chinese	32.27	17.40	30.06	34.85	38.72	51.24	56.82	37.34
watermarked bert-base-chinese (L_{wat} only)	28.50	13.20	15.98	23.48	31.08	34.43	42.98	27.09
contrastive learned bert-base-chinese (L_{con} only)	47.98	**42.86**	**52.22**	52.87	55.46	57.64	**62.13**	**53.02**
TMRW bert-base-chinese	**48.03**	37.51	51.73	**54.78**	**57.36**	**58.80**	59.03	52.46
bert-base-uncased	21.02	20.12	16.77	20.14	27.43	36.64	49.34	27.35
watermarked bert-base-uncased (L_{wat} only)	23.54	10.90	15.24	20.84	25.04	39.51	57.16	27.46
contrastive learned bert-base-uncased (L_{con} only)	26.02	**27.48**	24.00	26.92	31.74	39.23	55.42	32.97
TMRW bert-base-uncased	**29.83**	24.92	**34.51**	**34.68**	**36.97**	**40.06**	**61.65**	**37.52**

Table 2. Black-Box Verification (PCR) on Downstream Models.

Dataset	Base Model	PCR	
		Frozen	Unfrozen
AG News	TMRW bert-base-uncased	1.0	1.0
CoLA	TMRW bert-base-uncased	1.0	1.0
LCQMC	TMRW bert-base-chinese	1.0	1.0

watermark under two fine-tuning paradigms:

- **Full Fine-tuning (Unfrozen Encoder)**: All parameters of the watermarked encoder are updated during downstream task training. This represents the most challenging scenario for watermark survival.
- **Feature Extraction (Frozen Encoder)**: The parameters of the watermarked encoder are kept frozen, and only the newly added classification head is trained.

We assess robustness on three distinct classification tasks: AGnews, CoLA, and LCQMC. The downstream model M in our experiment is based on bert model and an additional layer as classification head and is fine-tuned on each downstream task for 10 epochs. Subsequently, X_{trig} is passed through the fine-tuned downstream model to evaluate the PCR. According to Equation (15), a PCR value approaching 1 indicates that the trigger inputs consistently produce a near-unanimous output, a statistically anomalous behavior confirming the watermark.

The black-box verification results are presented in Table 2. The results show that the watermark remains verifiable across all downstream models. As shown in Table 2, the PCR for X_{trig} is a perfect 1.0 in every single experiment. This outcome demonstrates the effectiveness of our watermarking scheme. The L_{wat} forces the trigger embeddings into a tight cluster in the feature space that these embeddings become indistinguishable for downstream classifiers. As a result, the classifier learns to map this entire cluster of embeddings to a single output class.

The success under both frozen and unfrozen conditions is noteworthy. Even when the encoder is unfrozen, PCR remains 1.0. It shows that the geometric property of X_{trig} persists through the fine-tuning process. These findings offer compelling evidence for the robustness of our TMRW framework.

Robustness against Fine-Tuning Attack

To further probe the limits of our watermark's robustness, we conduct a fine-tuning attack test. While the watermark survives our standard fine-tuning protocols (10 epochs), this experiment aims to identify its breaking point by subjecting the model to a prolonged, continuous training. This allows us to measure the watermark's persistence as a function of fine-tuning intensity.

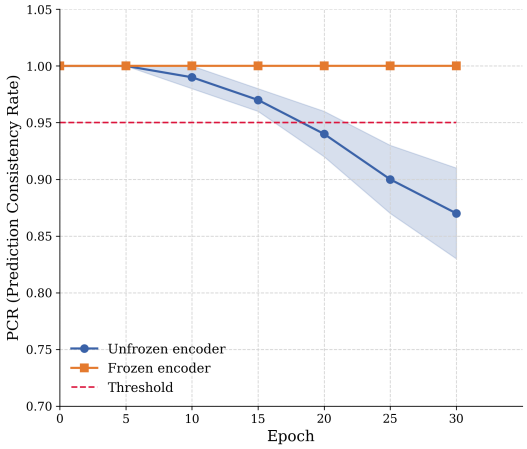


Figure 4. *Watermark durability during fine-tuning on the CoLA dataset. The plot tracks the PCR for full fine-tuning (blue) and frozen encoder (orange). The dashed line marks the successful verification threshold of 0.95.*

We take the fine-tuned downstream model based on TMRW bert-base-uncased and continuously train it on the CoLA dataset for an extended period of 30 epochs. We compare two scenarios: full fine-tuning where all parameters are updated (Unfrozen encoder), and a control scenario where the watermarked layers are frozen (Frozen encoder). After 5 epoch intervals, we save a model checkpoint and perform a black-box PCR verification. This methodology allows for a detailed tracking of the degradation of the watermark signal under fine-tuning attack. The results of this test are visualized in Figure 4.

The plot reveals that the watermark possesses remarkable durability. The frozen encoder experiment (orange line) confirms that when the parameters of watermarked layers are not trained, the PCR remains at a perfect 1.0. During the unfrozen encoder fine-tuning (blue line), the black-box PCR remains near-perfect for the initial epochs before commencing a gradual decline. Additionally, the PCR remains above the 0.95 success threshold for approximately 18 epochs.

This behavior highlights a property of our method: a gradual degradation rather than an abrupt failure. This suggests that the geometric structure of the trigger cluster in the embedding space

Table 3. Downstream task performance comparison between clean and TMRW-watermarked models. The performance drop (Δ) is consistently minimal across all tasks.(Accuracy/F1 in %, MCC × 100).

Task	Encoder Version	Metric	Score	Δ
AGnews	Clean Model	Acc.	84.5	-0.3
	Watermarked		84.2	
CoLA	Clean Model	MCC	60.1	-0.6
	Watermarked		59.5	
LCQMC	Clean Model	F1	91.2	-0.2
	Watermarked		91.0	

is so distinct that it requires substantial and prolonged parameter updates to fully erase it. This stress test provides a quantitative measure of the watermark's robustness against fine-tuning attack. Given that most fine-tuning procedures in practice converge well within 5-10 epochs, the persistence of our watermark for approximately 18 epochs demonstrates a level of robustness that exceeds the requirements of normal fine-tuning cases.

Impact on Downstream Task Performance

An ideal watermarking framework should impose minimal degradation on downstream task performance. To quantify this impact, we evaluate our downstream models based on TMRW-watermarked encoders on three distinct classification tasks and compare their performance against downstream models based on non-watermarked baseline encoders under identical conditions.

We fine-tune both the baseline (clean model) and the TMRW watermarked model on AGnews, CoLA, and LCQMC. The comprehensive performance comparison is presented in Table 3. The results demonstrate that our TMRW framework has little negative impact on downstream tasks.

As shown in Table 3, the performance difference between the watermarked models and clean models is consistently marginal. For instance, on the AG News topic classification task, the TMRW watermarked model achieves an accuracy of 84.2%, representing a mere 0.3 points drop compared to the baseline. Similarly, on the CoLA grammatical acceptability task, the drop in the MCC score is only 0.6 points. For the Chinese paraphrase task LCQMC, F1-score sees a minimal decrease of only 0.2 points.

These results confirm that the TMRW framework maintains the ability to embed a task migration resistant watermark while incurring a performance degradation of less than 1 point on diverse downstream tasks, making it a highly practical and deployable solution for model ownership protection.

Conclusion

In this paper, we address the fragility of existing watermarks against task migration. We introduce the TMRW framework, a novel approach that leverages a joint training paradigm combining a covariance-based watermark loss with a performance-preserving contrastive loss. Our comprehensive experiments validate the success of the TMRW framework across three fundamental dimensions: effectiveness, fidelity, and robustness. The effectiveness of the framework is confirmed through white-box and black-box verification. Critically, TMRW is achieved with high fidelity. Our joint training strategy not only avoids degrading the model's performance but also improves the encoder's semantic representation capabilities. Furthermore, TMRW demonstrate exceptional robustness, exhibiting resilience to task migration and a certain degree of fine-tuning attacks. Our TMRW framework provides an effective solution for protecting the ownership of pre-trained language encoder models. By ensuring the persistence of watermark through the entire model lifecycle, from pre-training to downstream deployment, our work takes a step towards building a more secure and trustworthy ecosystem for shared NLP resources.

Acknowledgment

This study was financially supported by the Nanning "Yong Jiang" Program under Grant Number RC20250102, Science and Technology Commission of Shanghai Municipality under Grant Number 24ZR1424000, and Xizang Autonomous Region Central Guided Local Science and Technology Development Fund Project under Grant Number XZ202401YD0015. This research was also partly supported by the National Natural Science Foundation of China under Grant Number U23B2023.

References

[1] C. Kong, J. Chen, S. Tan, Z. Yin, and X. Zhang. Copyright protection for large language model EaaS via unforgeable backdoor watermarking. *International Conference on Pattern Recognition*, pp. 1–15, 2024.

[2] Z. Wang, B. Wu, J. Deng, and Y. Yang. Robust and minimally invasive watermarking for EaaS. *Findings of the Association for Computational Linguistics*, pp. 2167–2191, 2025.

[3] G. Zhao and C. Qin. Black-box lossless fragile watermarking based on hidden space search for DNN integrity authentication. In: *Proc. Asia Pacific Signal and Information Processing Association Annual Summit and Conference*, pp. 450–455, 2023.

[4] Z. Fei, B. Yi, J. Geng, R. He, L. Nie, and Z. Liu. Your fixed watermark is fragile: towards semantic-aware watermark for EaaS copyright protection. *arXiv preprint arXiv:2411.09359v1*, 2024.

[5] H. Wu, G. Liu, Y. Yao, and X. Zhang. Watermarking neural networks with watermarked images. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 7, pp. 2591-2601, 2021.

[6] W. Peng, J. Yi, F. Wu, S. Wu, B. Wu, L. Lyu, B. Jiao, T. Xu, G. Sun, and X. Xie. Are you copying my model? Protecting the copyright of large language models for EaaS via backdoor watermark. In: *Proc. Annual Meeting of the Association for Computational Linguistics*, pp. 7653-7668, 2023.

[7] A. Shetty, Y. Teng, K. He, and Q. Xu. WARDEN: Multi-directional backdoor watermarks for embedding-as-a-service copyright protection. *arXiv preprint arXiv:2403.01472*, 2024.

[8] H. Li, Y. Ren, Y. Cao, Y. Li, F. Fang, and X. Wang. From essence to defense: Adaptive semantic-aware watermarking for embedding-as-a-service copyright protection. *arXiv preprint arXiv:2512.16439*, 2025.

[9] T. Zhang, H. Wu, X. Lu *et al.* AWEncoder: Adversarial watermarking pre-trained encoders in contrastive learning. *Applied Sciences*, vol. 13, no. 6, p. 3531, 2023.

[10] P. Khosla, P. Teterwak, C. Wang *et al.* Supervised contrastive learning. *Advances in neural information processing systems*, vol. 33, pp. 18661–18673, 2020.

[11] T. Gao, X. Yao, and D. Chen. Simcse: Simple contrastive learning

of sentence embeddings. *arXiv preprint arXiv:2104.08821*, 2021.

[12] E. Agirre, C. Banea, C. Cardie *et al.* SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 252–263, 2015.

[13] E. Agirre, C. Banea, C. Cardie *et al.* SemEval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation*, pages 81–91, 2014.

[14] E. Agirre, C. Banea, D. Cer *et al.* SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, pages 497–511. Association for Computational Linguistics, 2016.

[15] E. Agirre, D. Cer, M. Diab, and A. Gonzalez-Agirre. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 385–393, 2012.

[16] E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre, and W. Guo. *SEM 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics, Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, 2013.

[17] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, pages 1–14, 2017.

[18] M. Marelli, S. Menini, M. Baroni *et al.* A SICK cure for the evaluation of compositional distributional semantic models. In *International Conference on Language Resources and Evaluation*, pages 216–223, 2014.

[19] X. Liu, Q. Chen, C. Deng *et al.* LCQMC: A Large-scale Chinese Question Matching Corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1952–1962. Association for Computational Linguistics, 2018.

[20] A. Warstadt, A. Singh, and S. R. Bowman. Neural Network Acceptability Judgments. *arXiv preprint arXiv:1805.12471*, 2018.

[21] X. Zhang, J. Zhao, and Y. LeCun. Character-level Convolutional Networks for Text Classification. *Advances in Neural Information Processing Systems 28*, 2015.

[22] J. Devlin, M. W. Chang, K. Lee *et al.* Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171-4186, 2019.

[23] A. Conneau and D. Kiela. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*, 2018.

Author Biography

Yijia Xu received the B.S. degree in Data Science from Shanghai University, Shanghai, China, in 2024. He is currently pursuing the M.S. degree in signal and information processing from the School of Communication and Information Engineering, Shanghai University, Shanghai, China. His research interests include AI security.

Gejian Zhao received the B.S. degree in electronic engineering from Yangzhou University, Yangzhou, China, in 2019, and the M.S. degree in signal and information processing from the University of Shanghai for Science and Technology, Shanghai, China, in 2023. He is currently pursuing the Ph.D. degree with the School of Communication and Information Engineering, Shanghai University, Shanghai, China. His research interests include multimedia hashing and AI security.

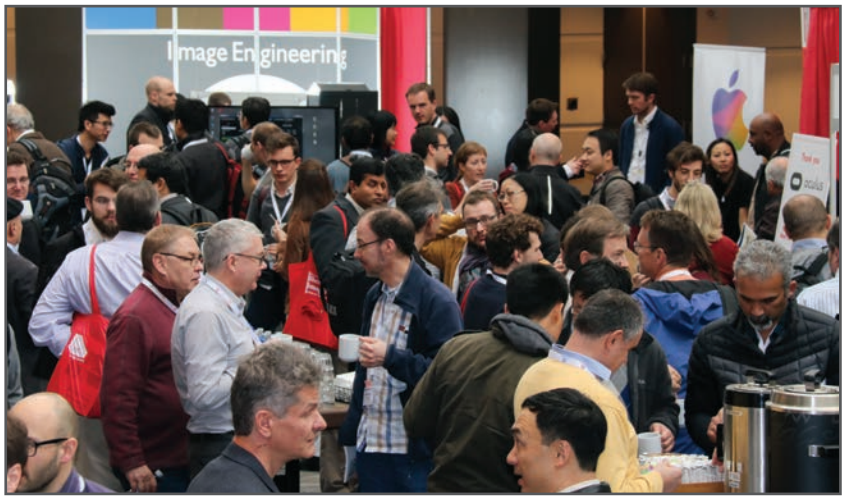
Hanzhou Wu received his BS and PhD degrees from Southwest Jiaotong University, Chengdu, China, in 2011 and 2017, respectively. He was a Visiting Scholar in New Jersey Institute of Technology, New Jersey, USA, from 2014 to 2016. He was a Research Scientist in Institute of Automation, Chinese Academy of Sciences, Beijing, China, from 2017 to 2019. He is now an Associate Professor in Shanghai University, Shanghai, China. His research interests include steganography, steganalysis, digital watermarking and digital forensics. He has published more than 100 research articles in peer journals and conferences. He has also written four book chapters. He served as the Organization Chair for 2022 IEEE International Workshop on Information Forensics and Security, and serves as an Associate Editor for IEEE Signal Processing Letters started from 2025.

Xinpeng Zhang received the B.S. degree in computational mathematics from Jilin University, Changchun, China, in 1995, and the M.E and Ph.D. degrees in communication and information system from Shanghai University, Shanghai, China, in 2001 and 2004, respectively. From 2010 to 2011,he was a Visiting Scholor with the State University of New York at Binghamton, Binghamton, NY, USA. From 2011 to 2012, he was an experienced Researcher with Konstanz University, Konstanz, Germany, sponsored by the Alexander von Humboldt Foundation. Since 2004, he has been with the faculty of the School of Communication and Information Engineering, Shanghai University, where he is a Professor. His research interests include multimedia security, AI security, and image processing. He has authored or coauthored more than 300 articles in these areas. From 2014 to 2017, he was an Associate Editor for the IEEE Transactions on Information Forensics and Security.

JOIN US AT THE NEXT EI!

electronic IMAGING

Imaging across applications . . . Where industry and academia meet!



• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

ADVANCING IMAGING SCIENCE 75 1947 IS&T 2022 imaging.org