Eguard: Defending LLM Embeddings Against Inversion Attacks via Text Mutual Information Optimization

Tiantian Liu[1,2,3], **Hongwei Yao**[1,2], **Feng Lin**[1,2*], **Tong Wu**[1,2], **Zhan Qin**[1,2], **Kui Ren**[1,2]

[1]State Key Laboratory of Blockchain and Data Security, Zhejiang University
[2]Hangzhou High-Tech Zone (Binjiang) Institute of Blockchain and Data Security
[3]School of Informatics, Xiamen University
{tiantian, yhongwei, flin, cocotwu, qinzhan, kuiren}@zju.edu.cn

Abstract

While text embeddings enable efficient semantic processing in LLMs, they remain vulnerable to inversion attacks that reconstruct sensitive original text. However, current defense methods typically treat text embeddings from the feature level independently, ignoring the exploitation of the mutual relation among the embedding construction pipeline. To address this limitation, we propose `Eguard`, a framework that effectively disrupts chains of relationships between the original semantic space and defended functional space. Our improvements manifest at two levels, i.e., the global-level and local-level mutual information. At the global level, we propose to minimize the statistical dependency between protected embeddings and their original inputs, effectively decoupling sensitive content from the semantic space accessible to adversaries. At the local level, we apply keyword-antonym contrastive learning to enforce semantic discriminability within the space of downstream utility. This synergy of global privacy control and local semantic alignment allows `Eguard` to achieve a superior privacy-utility trade-off than traditional defenses. Our approach significantly reduces privacy risks, protecting over 95 percent of tokens from inversion while maintaining high performance across downstream tasks consistent with original embeddings.

Extended version — https://arxiv.org/abs/2411.05034

Introduction

Large language models (LLMs) like ChatGPT, Claude (Anthropic 2024), and ChatGLM rely critically on embedding vector databases as their long-term memory(Topsakal and Akinci 2023; Ozdemir 2023), enabling retrieval-augmented generation (RAG) (Gan et al. 2024; Zhao et al. 2024) to enhance factual accuracy and overcome context window limitations. These dense vector representations power essential capabilities including semantic search, clustering, and context-aware generation, as demonstrated by widespread adoption in APIs like OpenAI's embeddings service. However, the richness of semantic information that makes embeddings valuable also creates privacy vulnerabilities.

Recent studies have revealed that embeddings are vulnerable to *inversion attacks* (Lin et al. 2024; Li, Xu, and Song 2023; Kale et al. 2025), where adversaries exploit embedding vectors to reconstruct sensitive input texts. Such attacks pose severe risks in scenarios involving proprietary data or personal information. Researchers have proposed several defense mechanisms to mitigate these risks. These defenses can generally be categorized into three main approaches: noise superposition (Wen, Yiu, and Hui 2021; Liu et al. 2024), perturbation-based methods (Jin et al. 2025), and differential privacy (DP)-based defenses (Wang et al. 2024; Li et al. 2024). Despite these efforts, existing defense methods face significant limitations that hinder their practical effectiveness against embedding inversion attacks. Defending against such attacks presents unique challenges: First, embedding vectors inherently encapsulate sensitive input features, making it difficult to disentangle and protect private information without severely disrupting their semantic integrity. Second, modifying embedding layers often compromises the accuracy of LLMs on downstream tasks. Given the versatility of LLMs across applications such as sentiment analysis, natural language inference, and text summarization, developing robust defense mechanisms that simul-

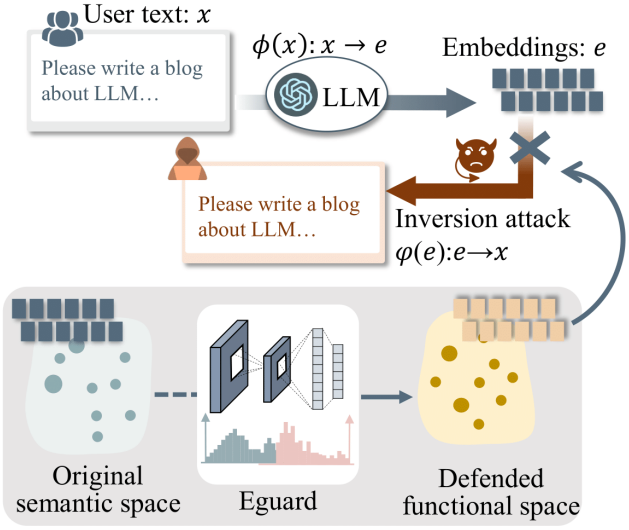


Figure 1: Overview of our defense against inversion attacks on embeddings.

*Corresponding author

taneously ensure privacy protection while preserving task-specific utility remains a critical research problem.

The embedding-to-text reconstruction process can be viewed as a Markov chain, and we mitigate inversion attacks by introducing an additional stage that disrupts the correlation between original text and reconstructed outputs. In this paper, we propose `Eguard` **(Embedding Guard)**, a novel defense mechanism that leverages mutual information optimization to shield embeddings from inversion attacks while preserving their effectiveness for natural language processing (NLP) tasks. The optimization process is driven by text mutual information, ensuring that the transformed embeddings retain task-relevant properties while minimizing their susceptibility to inversion. Specifically, `Eguard` introduces a projection network that acts as an intermediate stage between the original embedding and the reconstructed text, increasing the uncertainty of inversion attacks by leveraging information entropy principles. A fundamental challenge in this defense lies in decoupling sensitive information from projected embeddings, thereby significantly reducing the adversary's ability to recover original inputs. `Eguard` addresses this by leveraging global mutual information as a quantitative measure of relationships on the chain, systematically minimizing its approximation to sever the linkage between sensitive data and embeddings, thereby preventing information leakage. Additionally, protecting embeddings alone is not enough; their functional utility for downstream tasks must also be maintained. To address this, `Eguard` introduces a local mutual information optimization mechanism that enforces semantic discriminability through keyword-antonym contrastive learning. By maximizing the contrastive separation between keywords and their antonyms within a conditional distribution space, the model ensures that the projected embeddings preserve essential semantic distinctions while remaining anchored in the feature space necessary for robust task performance.

Our key contributions are summarized as follows:

- We propose `Eguard`, a transformer-based projection network that mitigates inversion attacks by projecting embeddings into a secure latent space while preserving downstream performance.
- We develop a mutual information-driven framework that simultaneously minimizes global information leakage and maintains local semantic discriminability, ensuring robust yet useful embeddings.
- We conduct comprehensive evaluations on seven embedding models and two ChatGPT embeddings under various attacks, demonstrating that `Eguard` blocks over 95% of token inversion, maintains more than 98% task consistency, and shows strong robustness to perturbations, unseen datasets, and adaptive attacks.

Embedding Models and Inversion Threats

An embedding model transforms an input text $x = [w_1, \ldots, w_l]$ from vocabulary $\mathcal{V}$ into a compact vector $e \in \mathbb{R}^d$, capturing semantic and syntactic features. Each token is mapped via $\boldsymbol{v}_i = \mathbf{W}(w_i)$ and processed by a Transformer encoder (e.g., BERT (Devlin et al. 2018), T5 (Raffel et al. 2020), LLaMA (Touvron et al. 2023)) to produce hidden states $\boldsymbol{h} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_l]$. A pooling operation (mean, max, or using the hidden state of a special token `[CLS]`) yields the final embedding $e = \text{Pooling}(\boldsymbol{h})$. These embeddings, serving as semantic representations, support applications in retrieval, classification, generation, and recommendation systems.

Adversary and Threat Model: Embedding inversion attacks attempt to reconstruct x from its embedding $\phi(x)$ by building an inverse model $\varphi(e) : e \rightarrow x$. We assume a strong adversary capable of acquiring embeddings (e.g., via database breaches) and equipped with an auxiliary dataset $\mathcal{D}_{aux}$ matching the distribution of target texts. An adaptive adversary may also exploit knowledge of defensive mechanisms to refine attack strategies. Given $\phi(x)$, the attacker seeks a candidate text $\hat{x}$ minimizing the distance to the original embedding:

$$\hat{x} = \arg\min_{\hat{x} \in \mathcal{V}} \|\phi(\hat{x}) - \phi(x)\|_2. \tag{1}$$

Brute-force search is computationally infeasible due to combinatorial explosion. To overcome this, attackers employ a decoder-based transformer φ (e.g., GPT-2) trained to maximize the likelihood:

$$\arg\max_{\varphi} \mathbb{E}_{x \sim \mathcal{D}_{aux}}[p(x|\varphi(\phi(x)))]. \tag{2}$$

The decoder generates tokens sequentially, modeling $p(x) = \prod_{i=1}^{n} p(t_i|\phi(x), t_{<i})$, with self-attention capturing long-range dependencies. A linear projection aligns embedding dimensions with the decoder input, enabling accurate text reconstruction from embeddings.

`Eguard` Defense

In this section, we present `Eguard`, a novel defense mechanism based on mutual information optimization, designed to mitigate embedding inversion attacks while preserving the functionality of embeddings. `Eguard` achieves this by projecting embedding vectors from their original space into a secured space, guided by two key objectives: global mutual information and local mutual information. The reduction of global mutual information increases the uncertainty of reconstructing original texts from inversion attacks, thereby detaching sensitive information and enhancing privacy. Simultaneously, the local mutual information between keywords and their corresponding antonyms ensures the preservation of discriminative semantic features, maintaining the effectiveness of embeddings for downstream tasks in large language models.

Global Mutual Information: Sensitive Feature Detachment

To effectively mitigate those inversion attacks, we propose an information-theoretic defense mechanism that reduces the dependency between original textual content and projected embeddings.

By analyzing the attack pipeline, we model the embedding generation and attacking process as a Markov chain:

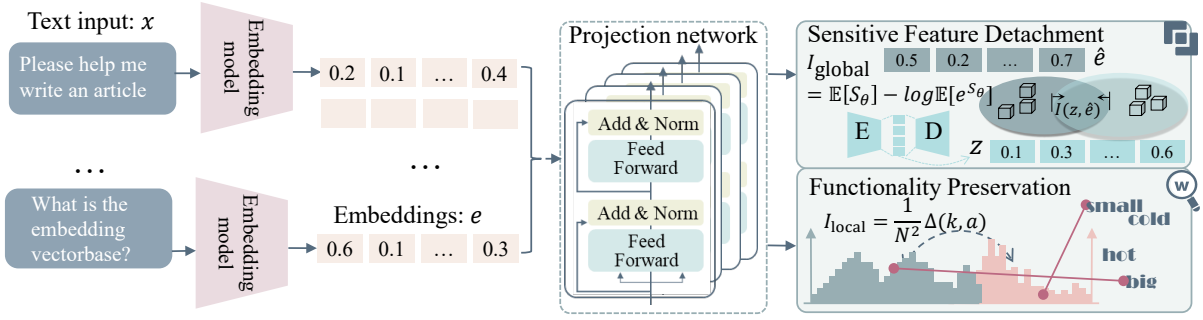


Figure 2: Overview of defense approach. `Eguard` contains a sensitive feature detachment module and a functionality preservation module.

$x \to e \to \hat{x}$, where x represents the original text, e is its corresponding embedding, and $\hat{x}$ is the reconstructed text obtained through the attack. To enhance security, we introduce a projection network, which transforms the original embedding e into a secured representation $\hat{e}$. Consequently, the Markov chain is extended as: $x \to e \to \hat{e} \to \hat{x}$. From an information-theoretic standpoint, the objective is to minimize the mutual information $I(x, \hat{e})$ between the original text x and the projected embedding $\hat{e}$, thereby reducing the amount of recoverable information:

The mutual information is formally defined through Shannon entropy relationships:

$$I(x, \hat{e}) = H(x) - H(x|\hat{e}), \tag{3}$$

where $H(\cdot)$ denotes Shannon entropy, and $H(x|\hat{e})$ represents the conditional entropy of x given $\hat{e}$. Prior research has established that mutual information can be approximated using the Kullback-Leibler (KL) divergence:

$$\begin{aligned} I(x, \hat{e}) &= \iint p(\hat{e}|x)p(x) \log \frac{p(x|\hat{e})}{p(\hat{e})} dx d\hat{e} \\ &= KL(p(\hat{e}|x)p(x) \| p(\hat{e})p(x)). \end{aligned} \tag{4}$$

Since x represents discrete text data, it is not directly possible to calculate the mutual information between x and a continuous vector $\hat{e}$. To address this, we introduce a pretrained autoencoder that is first trained on a large and unlabeled text corpus. The latent feature vector z produced by the autoencoder replaces x to estimate the mutual information $I(z, \hat{e})$. According to information theory, the transformation between z and x is lossless only if the autoencoder can perfectly reconstruct the input x from z. This implies that the conditional entropy $H(x|z)$ equals zero, ensuring that z can effectively substitute x in the calculation of mutual information.

Inspired by the work of Belghazi et al., we introduce a statistical network S parameterized by θ to estimate the joint probability density function of $\hat{e}$ and z. This enables us to derive an ideal approximation of $I(z, \hat{e})$, which can be formulated as:

$$\begin{aligned} I_{\text{global}}(z, \hat{e}) = &\sup_\theta \iint S_\theta(z, \hat{e}) p(z, \hat{e}) dz d\hat{e} \\ &- \log \left(\iint e^{S_\theta(z, \hat{e})} dp(z) dp(\hat{e}) \right). \end{aligned} \tag{5}$$

Eq. 5 provides a rigorous estimation of global mutual information, and its computational implementation is detailed in Algorithm 1. The training objective incorporating this global mutual information is defined as:

$$\mathcal{L} = \ell_{\text{task}} + \alpha I_{\text{global}}(z, \hat{e}), \tag{6}$$

where ℓ_{task} represents the loss function associated with the downstream task, and α is a hyperparameter that controls the influence of global mutual information regularization.

Lemma 1 *Minimizing the mutual information $I(x, \hat{e})$ between the original text x and the secured embedding $\hat{e}$ effectively protects against inversion attacks by reducing the recoverable information about x from the attacker's reconstructed text $\hat{x}$.*

Proof of Lemma 1: see Appendix A in the extended version for proof.

Lemma 2. *For any $\epsilon > 0$, there exists a neural network S_θ such that:*

$$|I(z, \hat{e}) - I_{\text{global}}(z, \hat{e})| \leq \epsilon, \quad \textit{almost surely}. \tag{7}$$

This lemma demonstrates that the global mutual information estimation $I_{\text{global}}(z, \hat{e})$ converges to the neural-based mutual information measure $I(z, \hat{e})$ as the number of training samples tends to infinity, ensuring the approximation error is bounded by ϵ with probability one.(Belghazi et al. 2018)

Proof of Lemma 2: see Appendix B in the extended version for proof.

Local Mutual Information: Functionality Preservation

While global mutual information ensures robustness against inversion attacks, it introduces a crucial challenge, that is, preserving the effectiveness of embeddings for downstream tasks. To address this, we propose local mutual information as a mechanism for functionality preservation. The methodology begins by decomposing the input text into a set of keywords and retrieving their corresponding antonyms, by leveraging NLTK toolkits in Python for automated extraction. The keywords k capture essential semantics and the antonyms a represent opposite context. These pairs (k, a) form the basis for contrastive learning, where the objective is to minimize the mutual information $I(k, a)$ between keywords and their antonyms. Formally, the upper bound on

Algorithm 1: Global and Local Mutual Information Optimization

Require: Original embeddings e, textual input x, pretrained autoencoder, projection network g_p, embedding encoder, statistical network S, and variational network V.

Ensure: Optimized projection network g_p and refined embeddings $\hat{e}$.

Initialize:
Encode input text into latent representations:
$z = \text{Autoencoder}(x)$
Extract keywords and antonyms:
$k, a = \text{Encoder}(\text{NLTK}(x))$
for each training iteration **do**
  $k \leftarrow g_p(k), a \leftarrow g_p(a)$;
  Estimate global mutual information:
  $I_{\text{global}} = \frac{1}{N}\sum_{i=1}^{N} S(g_p(e_i), z_i) - \log\left(\frac{1}{N}\sum_{i=1}^{N} e^{S(g_p(e_i), z_i)}\right)$;
  Estimate local mutual information:
  $I_{\text{local}} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}[\log V(a_i, k_i) - \log V(a_j, k_i)]$
  Update statistical network parameters:
  $\theta_S \leftarrow \theta_S + \eta\nabla_{\theta_S} I_{\text{global}}$
  $\theta_V \leftarrow \theta_V + \eta\nabla_{\theta_V}\frac{1}{N}\sum_{i=1}^{N}\log V(a_i, k_i)$;
  Optimize the projection network:
  $g_p \leftarrow g_p - \eta\nabla_{g_p}(\ell_{\text{task}} + \alpha I_{\text{global}} + \beta I_{\text{local}})$
end for

their mutual information is given by:

$$I_{\text{local}}(k,a) = \iint p(k,a)\log p(k|a), dk, da - \int p(k)\left[\int p(a)\log p(a|k), da\right] dk. \tag{8}$$

Lemma 3. *For any keyword-antonym distribution, $I(k,a) \leq I_{\text{local}}(k,a)$, with equality if and only if k and a are independent.* (Cheng et al. 2020)

Proof of Lemma 3: see Appendix C in the extended version for proof.

The practical estimation of $I_{\text{local}}(k,a)$ involves training a variational network $V_\theta(k,a)$ to approximate $\log p(k|a)$, implemented via contrastive learning. For N samples $\{(k_i, a_i)\}$, the objective becomes:

$$I_{\text{local}} = -\beta\mathbb{E}\left[\log\frac{e^{V_\phi(k_i,a_i)}}{\int e^{V_\theta(k_i,a)}p(a)\,da}\right] \approx -\beta\frac{1}{N^2}\sum_{i,j}\left[\log V_\theta(k_i,a_i) - \log V_\theta(k_i,a_j)\right], \tag{9}$$

where $p(a_i|k_i)$ represents the conditional log-likelihood of a positive sample pair (k_i, a_i) and $p(a_j|k_i)$ represents the conditional log-likelihood of a negative sample pair (k_i, a_j). The difference between the two terms reflects a contrastive probability log-ratio, ensuring that the model effectively distinguishes between meaningful keyword-antonym pairs. The local mutual information guarantees that the embeddings retain discriminative semantic features.

The unified training objective integrates global privacy protection and local utility preservation:

$$\mathcal{L} = \ell_{\text{task}}(y, f(g_p(e))) + \alpha I_{\text{global}}(z, \hat{e}) + \beta I_{\text{local}}(k, a). \tag{10}$$

where task-specific heads $f(\cdot)$ adapt to diverse applications: cross-entropy loss for classification, multiple negatives ranking loss (MNRL) for retrieval, and sequence cross-entropy for text generation. This formulation theoretically guarantees that sanitized embeddings $\hat{e}$ maintain sufficient statistics for downstream tasks while obfuscating extraneous information vulnerable to inversion attacks. The proposed Algorithm 1 systematically enhances embedding security while preserving their functional utility. The algorithm begins by encoding the input text into latent representations and extracting keywords and their antonyms. During each training iteration, the global mutual information is estimated using the statistical network S, while the local mutual information is computed via the variational network V. The parameters of S and V are updated to refine their estimates, and the projection network is optimized using gradient-based updates to balance privacy protection and task performance. This iterative process ensures that the embeddings are both secure and effective for downstream tasks. The projection network g_p is implemented using a 24-layer RoBERTa encoder, which serves as an intermediate transformation module between raw embeddings and the secured space.

Evaluation

Experimental Setup

Datasets and tasks. We evaluate on four representative text understanding tasks: sentiment analysis SST, natural language inference (NLI), question retrieval (QR), and text summarization (TS). Each dataset is encoded by the embedding models for classification, retrieval, or summarization. Details of the datasets are provided in Appendix E of the extended version.

Embedding model and Attack model. We use five models: T5, RoBERTa, MPNet, LLaMA, and Gemma, all with frozen parameters. T5 and MPNet produce 768-dimensional embeddings, while RoBERTa outputs 1024-dimensional vectors. Model details are in Appendix D of the extended version . Some additional results for Table 1 and Table 2 are included in the extended version due to space limits. We adopt GPT-2 as the inversion decoder, with a maximum sequence length of 128. The decoder is trained on in-domain data using Adamax with a learning rate of $2\text{e}{-2}$ and a batch size of 16.

Metrics. Defense performance is measured using F1, Recall, and BLEU, while task utility is assessed via Accuracy for SST, NLI, and QR, and ROUGE for TS. Lower F1, Recall, and BLEU indicate stronger protection, while higher task accuracy reflects better preservation of utility.

Overall Performance

Defense against embedding inversion attacks. We evaluate the effectiveness of our defense across five embedding models, including T5, RoBERTa, MPNet, LLaMA and

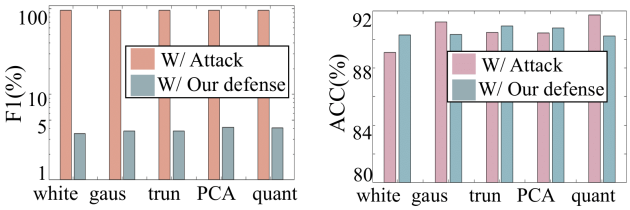
Model	Method	SST2			NLI			QR			TS		
		F1(%)	Recall(%)	BLEU	F1(%)	Recall(%)	BLEU	F1(%)	Recall(%)	BLEU	F1(%)	Recall(%)	BLEU
T5	W/ Attack	93.9	93.3	0.836	96.5	95.0	0.789	98.2	97.9	0.976	95.2	94.7	0.901
	FGSM	14.2	16.1	0.092	25.9	19.4	0.121	36.3	34.4	0.230	39.3	38.6	0.245
	FreeLB	39.8	39.4	0.433	42.2	44.9	0.268	46.4	43.6	0.278	49.0	48.8	0.312
	DPforward	9.35	11.9	0.054	23.2	17.4	0.139	21.7	16.4	0.089	21.5	20.3	0.100
	Sanitization	6.75	11.0	0.030	23.2	16.3	0.095	23.5	21.7	0.103	22.6	22.1	0.092
	Ours	**4.75**	**4.40**	**0.019**	**5.35**	**4.47**	**0.034**	**3.57**	**4.14**	**0.014**	**3.56**	**4.44**	**0.011**
LLaMA2	W/ Attack	93.9	93.1	0.831	83.3	81.1	0.948	98.5	98.1	0.985	96.9	95.9	0.914
	FGSM	14.2	16.1	0.092	43.2	34.5	0.352	37.9	36.6	0.237	38.8	37.3	0.218
	FreeLB	44.3	43.6	0.446	41.1	34.2	0.351	50.6	49.6	0.289	47.1	46.9	0.283
	DPforward	12.2	13.0	0.058	25.4	21.9	0.115	25.7	24.3	0.121	22.0	22.5	0.108
	Sanitization	11.9	13.3	0.108	24.3	20.4	0.125	23.7	22.9	0.134	25.2	24.9	0.142
	Ours	**5.63**	**4.97**	**0.014**	**4.43**	**3.18**	**0.009**	**4.13**	**3.29**	**0.010**	**3.53**	**4.12**	**0.011**
LLaMA3-70B	W/ Attack	96.5	94.2	0.924	91.0	88.0	0.856	96.8	97.6	0.958	98.6	95.6	0.923
	FGSM	19.4	20.3	0.117	32.3	27.3	0.215	37.3	35.6	0.234	35.6	32.7	0.194
	FreeLB	27.0	25.9	0.205	38.3	37.6	0.286	47.3	44.8	0.362	48.1	46.6	0.315
	DPforward	9.45	10.2	0.082	20.5	18.3	0.143	23.2	22.7	0.102	22.5	21.7	0.101
	Sanitization	11.9	13.7	0.103	19.3	17.7	0.085	24.0	23.7	0.120	27.0	26.4	0.108
	Ours	**4.67**	**5.06**	**0.007**	**4.01**	**4.23**	**0.007**	**4.07**	**3.75**	**0.010**	**4.89**	**4.97**	**0.005**
Gemma2-9B	W/ Attack	95.8	92.5	0.869	84.7	81.2	0.830	90.7	86.8	0.870	97.2	96.9	0.950
	FGSM	15.6	17.1	0.101	33.4	31.3	0.219	36.8	34.7	0.224	34.7	33.9	0.222
	FreeLB	33.8	28.8	0.221	41.8	40.9	0.265	53.7	50.1	0.372	49.9	48.2	0.314
	DPforward	9.54	10.7	0.072	20.0	19.8	0.128	20.7	19.5	0.113	19.9	17.6	0.099
	Sanitization	9.62	10.9	0.067	20.9	17.9	0.095	24.1	23.8	0.109	22.1	20.5	0.094
	Ours	**4.20**	**4.50**	**0.007**	**4.38**	**4.42**	**0.012**	**4.81**	**4.63**	**0.009**	**4.50**	**4.60**	**0.012**

Table 1: The overall performance of Eguard and other defense against embedding inversion attacks. Model: the type of embedding models, SST2, NLI, QR, and TS are the corresponding downstream datasets. Extended results are reported in the Appendix F of the extended version.

Gemma, and compare it with adversarial training methods FreeLB (Zhu et al. 2019) and FGSM (Kim 2020), as well as differential privacy approaches DPSanitization (Tong et al. 2024) and DPForward (Du et al. 2023; Zhang et al. 2025). The baseline row in Table 1, labeled W attack, highlights the vulnerability of embeddings where inversion success rates exceed 95%. FreeLB and FGSM provide only partial protection, with F1 and Recall dropping to 15–50%, due to their perturbations failing to fundamentally reshape the semantic space exploited by attackers. In contrast, differential privacy achieves stronger robustness with F1 and Recall between 9% and 29% but often reduces the quality of embeddings for downstream tasks. Our defense significantly outperforms these methods, lowering inversion success to roughly 4% and protecting over 95 percent of tokens, demonstrating a more substantial disruption of attacker inference.

Evaluation on harmlessness. To determine whether privacy protection compromises task performance, we evaluate the defended embeddings on sentiment analysis, natural language inference, question retrieval and text summarization. The results 2 reveal that while differential privacy methods reduce downstream accuracy due to heavy noise injection, and adversarial training slightly improves robustness but still lags behind, our approach retains over 98% of the original accuracy and ROUGE scores. These results indicate that our method successfully balances security and utility by decoupling sensitive information from the semantic representations without impairing discriminative power.

Defense overhead. We measure the training and inference overhead introduced by our projection network on T5 and MPNet using two NVIDIA RTX A6000 GPUs. On average, our method incurs a one-time training overhead of 1.6x–2.4x for MPNet and 2.1x–3.4x for T5 compared to undefended models (e.g., 16.3ms vs. 9.6ms per batch for SST2). Despite this additional cost, the overhead is acceptable given the substantial improvement in security and privacy.



(a) Defense results. (b) Downstream performance.

Figure 3: The defense performance and downstream task performance under embedding perturbations.

Evaluation on Robustness

Robustness to embedding perturbations. To evaluate robustness against typical perturbations in cloud storage or transmission, we add white noise, Gaussian noise, truncation, PCA and quantization to T5 embeddings on the SST2 dataset. As shown in Figure 3, noise slightly reduces downstream performance due to semantic loss, while PCA and

Model	Method	SST2(%)	NLI(%)	QR(%)	TS(%)
T5	W/O Attack	**94.3**	**81.4**	**96.9**	**39.6**
	FGSM	82.7	77.1	80.3	25.8
	FreeLB	84.4	80.1	84.4	28.6
	DPforward	60.7	46.9	48.9	19.4
	Sanitization	54.6	46.7	50.2	18.9
	Ours	**93.8**	**81.8**	**96.7**	**38.3**
LLaMA2	W/O Attack	**97.1**	**82.6**	**98.9**	**39.6**
	FGSM	78.7	77.4	80.2	25.4
	FreeLB	82.1	72.3	83.1	20.3
	DPforward	60.8	56.9	60.1	18.9
	Sanitization	51.9	40.6	50.1	18.3
	Ours	**96.8**	**81.8**	**98.1**	**38.7**
LLaMA3-70B	W/O Attack	**95.2**	**82.7**	**99.8**	**40.1**
	FGSM	84.5	77.0	72.1	24.2
	FreeLB	81.8	77.5	74.1	22.3
	DPforward	71.1	69.6	63.1	20.1
	Sanitization	69.3	72.8	58.2	18.9
	Ours	**94.8**	**80.3**	**97.9**	**39.8**
Gemma2-9B	W/O Attack	**93.4**	**80.8**	**96.3**	**38.9**
	FGSM	85.8	73.4	82.7	23.1
	FreeLB	79.4	65.7	73.5	20.4
	DPforward	74.7	55.2	67.7	17.9
	Sanitization	63.5	51.4	63.7	17.7
	Ours	**93.4**	**80.8**	**95.8**	**38.6**

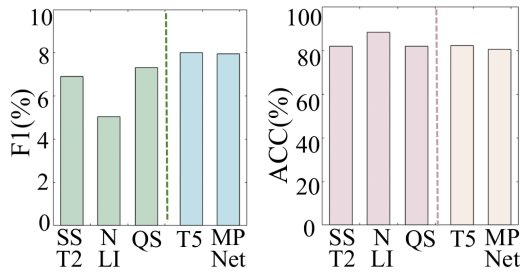
Table 2: Harmlessness evaluation of original and defended embeddings. ROUGE for summarization, accuracy for others. More results in Appendix G of the extended version

Model	MPNet			T5		
Dataset	SST2	NLI	TS	SST2	NLI	TS
W/O Defense	9.6ms	10.4ms	14.8ms	6.3ms	12.9ms	14.9ms
W/ Defense	16.3ms	25.1ms	24.7ms	21.1ms	34.2ms	31.6ms
Δ	6.7ms	14.7ms	9.9ms	14.8ms	21.3ms	16.7ms

Table 3: Overhead training comparison of `Eguard` per training batch versus the original undefended training in the same setting.

truncation preserve stable results thanks to the redundancy of high-dimensional features. Under quantization, our defense still prevents more than 95 percent of token inversion and maintains over 89% downstream accuracy, indicating that such perturbations do not significantly weaken the protected space.

Generalization to defenses unseen during training. We examine generalization by training the projection network on one dataset or embedding model and evaluating on unseen ones. As shown in Figure 4, the left side of each subfigure presents results under the unseen dataset setting (e.g., training on NLI, testing on SST2 and QR), while the right shows the unseen embedding model setting (e.g., training on T5, testing on MPNet). The result on unseen datasets shows that using NLI for training yields better cross-dataset robustness due to its larger data scale. When transferring across embedding models such as training on T5 and testing on MPNet, both defense performance and task accuracy drop because of distinct feature spaces, yet our method still surpasses differential privacy baselines in maintaining both security and utility.



(a) Defense performance. (b) Task performance.

Figure 4: Robustness under unseen conditions.

Attacking decoder. We further replace GPT-2 with LLaMA2-7B, LLaMA3-8B and Gemma2-9B as inversion decoders targeting SST2 embeddings from T5, LLaMA2-7B and Gemma2-9B. Table 4 shows that all decoders display similar attack performance, but our defense consistently mitigates inversion attempts, demonstrating resilience across different large language model architectures.

Model	Defense	Attacking Decoder (F1%)		
		LLaMA2	LLaMA3	Gemma2
T5	W/O	97.2	98.4	95.8
	W/	3.70	3.91	4.03
LLaMA2	W/O	92.9	94.4	91.8
	W/	4.10	4.04	3.91
Gemma2	W/O	93.5	98.2	98.6
	W/	4.20	4.01	4.04

Table 4: The performance under different attacking decoders

OpenAI embeddings

We evaluate embedding inversion attacks and defense strategies on OpenAI embeddings, which are also widely used for clustering, retrieval, and RAG applications. We use the OpenAI API with texts from the SST2 dataset to query three public models: text-embedding-3-small, text-embedding-3-large, and text-embedding-ada-002, with embedding dimensions of 1536, 1536, and 3072, respectively. Table 5 summarizes the performance of our defense compared to other baselines. Undefended embeddings show high F1 and Recall, confirming their vulnerability to inversion attacks. FGSM and FreeLB achieve limited protection, while DPForward and Sanitization offer moderate improvements. In contrast, our method achieves the lowest attack success rates, with F1 reduced to 5.28% for ada-002, 5.12% for 3-small, and 3.88% for 3-large, and Recall values ranging from 3.96% to 4.73%. These results demonstrate that our defense provides robust protection while maintaining strong embedding utility, outperforming all other approaches.

Method	ada-002(%)		3-small(%)		3-large(%)	
	F1	Recall	F1	Recall	F1	Recall
W/ Attack	93.91	93.13	93.83	93.07	83.92	82.17
FGSM	13.28	11.37	13.54	11.37	13.41	10.26
FreeLB	14.19	12.11	13.95	11.06	14.57	10.19
DPforward	14.20	9.97	14.57	18.89	18.17	18.81
Sanitization	13.18	9.62	12.07	12.94	12.96	10.46
Ours	5.28	4.72	5.12	4.73	3.88	3.96

Table 5: Defense performance against embedding inversion attacks on OpenAI embeddings

Ablation Studies

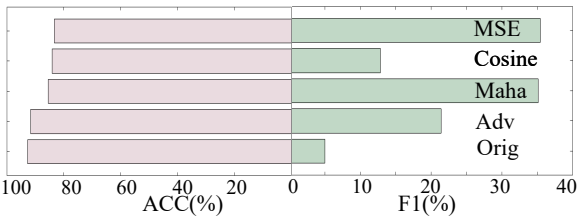
To evaluate the role of each component in our defense framework, we conduct ablation experiments by removing or modifying the mutual information loss and the projection network. The results are summarized in Figure 5.

Impact of loss function. We compare mutual information loss with mean squared error (MSE), cosine similarity, Mahalanobis distance, and adversarial loss. MSE and Mahalanobis losses reduce downstream accuracy to below 86%, and adversarial loss reaches about 92% but provides limited defense. Cosine similarity performs slightly better yet remains inferior to mutual information loss. The latter effectively reduces statistical dependency between original and protected embeddings, resulting in both stronger resistance to inversion and higher task performance. This confirms that capturing and minimizing information flow is more effective than merely optimizing distance-based metrics.

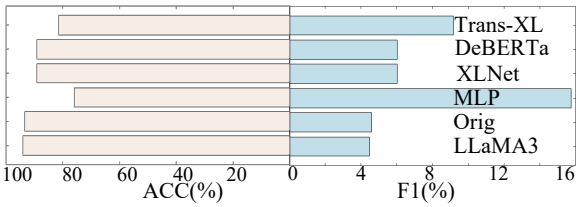
Impact of projection network. We further examine the influence of projection network architecture by comparing our 24-layer RoBERTa projection with LLaMA3, Transformer-XL, DeBERTa, XLNet, and a shallow MLP on SST2. While MLP fails to provide robust protection or maintain accuracy due to limited representational capacity, transformer-based architectures achieve stronger defense and comparable downstream performance. Deeper models like LLaMA3 and DeBERTa exhibit better generalization, suggesting that network depth and attention mechanisms are crucial for capturing complex semantic relations while introducing sufficient transformation to obscure sensitive content. Overall, our projection network strikes a balance between privacy protection and utility that simpler models cannot match.

Related Work

Text embeddings are universal: Text embeddings represent words, phrases, or documents as low-dimensional vectors, capturing semantic information for diverse NLP tasks (Ashkboos et al. 2024; Wang et al. 2023; Feng et al. 2020). Models such as Sentence-BERT (Reimers and Gurevych 2019), SimCSE (Gao, Yao, and Chen 2021), and Sentence-T5 (Ni et al. 2021) fine-tune pre-trained encoders to generate embeddings for classification, QA, retrieval, and bi-text mining. Recent efforts also extend embeddings across multiple domains, such as C-Pack for Chinese embeddings (Xiao et al. 2023), OpenAI embeddings for text and code (Neelakantan et al. 2022), and BGE for multilingual retrieval (Luo et al. 2024). Benchmark suites (MTEB (Muennighoff et al. 2022), SentEval (Conneau and Kiela 2018), BEIR (Thakur et al. 2021)) help compare embedding models on cross-domain tasks. Embeddings also enable anonymized storage of semantic data, facilitating model fine-tuning and personalized applications, as adopted by JINA AI (AI 2024), SingleStore (SingleStore 2023), and LangChain.



(a) Impact of loss function.



(b) Impact of projection network.

Figure 5: Ablation study results: (1) adjusting the loss function, and (2) investigating the impact of projection network architecture.

Text embeddings security: Embedding vectors are vulnerable to adversarial, membership inference, and reconstruction attacks (Song and Raghunathan 2020; Abdalla et al. 2020; Morris et al. 2023; Li, Xu, and Song 2023; Gu et al. 2023), which can expose sensitive information. To mitigate these risks, two main strategies exist: i) Differential privacy: InferDPT and DP-zero (Zhang et al. 2023; Tong et al. 2025) injects noise into gradients during optimization. Variants like d-privacy (Feyisetan et al. 2020) or DP-Forward (Du et al. 2023) perturb embeddings directly with controlled Gaussian noise. ii) Adversarial training: Models are optimized to resist adversarial perturbations (Liu et al. 2023; Wang et al. 2021), often improving the privacy-accuracy trade-off. Dai et al. introduce interpretable perturbations in the embedding space (Dai et al. 2019), while Yang et al. employ fast triplet metric learning to generate robust embeddings (Yang, Wang, and He 2022).

Conclusion

In this paper, we propose `Eguard`, a novel defense method to achieve superior privacy-utility balance through dual-level mutual information optimization, addressing the critical issue of embedding inversion attacks. `Eguard` simultaneously minimizing global mutual information to prevent sensitive information leakage, while preserving task-relevant semantics through local contrastive learning. Extensive experiments across seven embedding models and four downstream tasks demonstrate that `Eguard` consistently achieves strong resistance to inversion with minimal impact on performance.

Acknowledgments

The authors would like to thank our chairs and all the anonymous reviewers for their insightful comments. This work was supported in part by the National Key R&D Program of China under Grant 2023YFB2904000 and 2023YFB2904001, in part by the National Natural Science Foundation of China under Grant U2436206, 62032021, 62372406, and in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LZ25F020005.

References

Abdalla, M.; Abdalla, M.; Hirst, G.; and Rudzicz, F. 2020. Exploring the privacy-preserving properties of word embeddings: algorithmic validation study. *Journal of medical Internet research*, 22(7): e18055.

AI, J. 2024. Choosing the Right Embeddings. https://jina.ai/embeddings/.

Anthropic, A. 2024. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1.

Ashkboos, S.; Croci, M. L.; Nascimento, M. G. d.; Hoefler, T.; and Hensman, J. 2024. Slicegpt: Compress large language models by deleting rows and columns. *arXiv preprint arXiv:2401.15024*.

Belghazi, M. I.; Baratin, A.; Rajeshwar, S.; Ozair, S.; Bengio, Y.; Courville, A.; and Hjelm, D. 2018. Mutual information neural estimation. In *International conference on machine learning*, 531–540. PMLR.

Cheng, P.; Hao, W.; Dai, S.; Liu, J.; Gan, Z.; and Carin, L. 2020. Club: A contrastive log-ratio upper bound of mutual information. In *International conference on machine learning*, 1779–1788. PMLR.

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

Dai, Q.; Shen, X.; Zhang, L.; Li, Q.; and Wang, D. 2019. Adversarial training methods for network embedding. In *The world wide web conference*, 329–339.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Du, M.; Yue, X.; Chow, S. S.; Wang, T.; Huang, C.; and Sun, H. 2023. Dp-forward: Fine-tuning and inference on language models with differential privacy in forward pass. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2665–2679.

Feng, F.; Yang, Y.; Cer, D.; Arivazhagan, N.; and Wang, W. 2020. Language-agnostic BERT sentence embedding. *arXiv preprint arXiv:2007.01852*.

Feyisetan, O.; Balle, B.; Drake, T.; and Diethe, T. 2020. Privacy-and utility-preserving textual analysis via calibrated multivariate perturbations. In *Proceedings of the 13th international conference on web search and data mining*, 178–186.

Gan, C.; Yang, D.; Hu, B.; Zhang, H.; Li, S.; Liu, Z.; Shen, Y.; Ju, L.; Zhang, Z.; Gu, J.; et al. 2024. Similarity is Not All You Need: Endowing Retrieval Augmented Generation with Multi Layered Thoughts. *arXiv preprint arXiv:2405.19893*.

Gao, T.; Yao, X.; and Chen, D. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Gu, K.; Kabir, E.; Ramsurrun, N.; Vosoughi, S.; and Mehnaz, S. 2023. Towards sentence level inference attack against pre-trained language models. *Proceedings on Privacy Enhancing Technologies*.

Jin, S.; Pang, X.; Wang, Z.; Wang, H.; Du, J.; Hu, J.; and Ren, K. 2025. Safeguarding LLM Embeddings in End-Cloud Collaboration via Entropy-Driven Perturbation. *arXiv preprint arXiv:2503.12896*.

Kale, K.; Mylonakis, K.; Roberts, J.; and Roy, S. 2025. BeamClean: Language Aware Embedding Reconstruction. *arXiv preprint arXiv:2505.13758*.

Kim, H. 2020. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*.

Li, H.; Xu, M.; and Song, Y. 2023. Sentence Embedding Leaks More Information than You Expect: Generative Embedding Inversion Attack to Recover the Whole Sentence. In Rogers, A.; Boyd-Graber, J. L.; and Okazaki, N., eds., *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, 14022–14040. Association for Computational Linguistics.

Li, X.; Liu, W.; Lou, J.; Hong, Y.; Zhang, L.; Qin, Z.; and Ren, K. 2024. Local differentially private heavy hitter detection in data streams with bounded memory. *Proceedings of the ACM on Management of Data*, 2(1): 1–27.

Lin, Y.; Zhang, Q.; Cai, Q.; Hong, J.; Ye, W.; Liu, H.; and Duan, B. 2024. An inversion attack against obfuscated embedding matrix in language model inference. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2100–2104.

Liu, X.; Dai, S.; Fiumara, G.; and De Meo, P. 2023. An adversarial training method for text classification. *Journal of King Saud University-Computer and Information Sciences*, 35(8): 101697.

Liu, Z.; Wang, W.; Liang, H.; and Yuan, Y. 2024. Enhancing data utility in personalized differential privacy: A fine-grained processing approach. In *International Conference on Data Security and Privacy Protection*, 47–66. Springer.

Luo, K.; Liu, Z.; Xiao, S.; and Liu, K. 2024. BGE Landmark Embedding: A Chunking-Free Embedding Method For Retrieval Augmented Long-Context Large Language Models. *arXiv preprint arXiv:2402.11573*.

Morris, J. X.; Kuleshov, V.; Shmatikov, V.; and Rush, A. M. 2023. Text embeddings reveal (almost) as much as text. *arXiv preprint arXiv:2310.06816*.

Muennighoff, N.; Tazi, N.; Magne, L.; and Reimers, N. 2022. MTEB: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.

Neelakantan, A.; Xu, T.; Puri, R.; Radford, A.; Han, J. M.; Tworek, J.; Yuan, Q.; Tezak, N.; Kim, J. W.; Hallacy, C.; et al. 2022. Text and code embeddings by contrastive pretraining. *arXiv preprint arXiv:2201.10005*.

Ni, J.; Abrego, G. H.; Constant, N.; Ma, J.; Hall, K. B.; Cer, D.; and Yang, Y. 2021. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv preprint arXiv:2108.08877*.

Ozdemir, S. 2023. *Quick start guide to large language models: strategies and best practices for using ChatGPT and other LLMs*. Addison-Wesley Professional.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67.

Reimers, N.; and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

SingleStore. 2023. Using SingleStoreDB ChatGPT for Custom Data Sets. https://www.singlestore.com/blog/using-singlestoredb-chatgpt-for-custom-data-sets/.

Song, C.; and Raghunathan, A. 2020. Information leakage in embedding models. In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, 377–390.

Thakur, N.; Reimers, N.; Rücklé, A.; Srivastava, A.; and Gurevych, I. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*.

Tong, M.; Chen, K.; Yuan, X.; Liu, J.; Zhang, W.; Yu, N.; and Zhang, J. 2024. On the Vulnerability of Text Sanitization. *arXiv preprint arXiv:2410.17052*.

Tong, M.; Chen, K.; Zhang, J.; Qi, Y.; Zhang, W.; Yu, N.; Zhang, T.; and Zhang, Z. 2025. Inferdpt: Privacy-preserving inference for black-box large language models. *IEEE Transactions on Dependable and Secure Computing*.

Topsakal, O.; and Akinci, T. C. 2023. Creating large language model applications utilizing langchain: A primer on developing llm apps fast. In *International Conference on Applied Engineering and Natural Sciences*, 1050–1056.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Wang, L.; Yang, N.; Huang, X.; Yang, L.; Majumder, R.; and Wei, F. 2023. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*.

Wang, N.; Wang, S.; Li, M.; Wu, L.; Zhang, Z.; Guan, Z.; and Zhu, L. 2024. Balancing differential privacy and utility: A relevance-based adaptive private fine-tuning framework for language models. *IEEE Transactions on Information Forensics and Security*.

Wang, X.; Yang, Y.; Deng, Y.; and He, K. 2021. Adversarial training with fast gradient projection method against synonym substitution based text attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Wen, J.; Yiu, S.-M.; and Hui, L. C. 2021. Defending against model inversion attack by adversarial examples. In *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, 551–556. IEEE.

Xiao, S.; Liu, Z.; Zhang, P.; and Muennighof, N. 2023. C-pack: Packaged resources to advance general chinese embedding. *arXiv preprint arXiv:2309.07597*.

Yang, Y.; Wang, X.; and He, K. 2022. Robust textual embedding against word-level adversarial attacks. In *Uncertainty in Artificial Intelligence*, 2214–2224. PMLR.

Zhang, L.; Li, B.; Thekumparampil, K. K.; Oh, S.; and He, N. 2023. Dpzero: Private fine-tuning of language models without backpropagation. *arXiv preprint arXiv:2310.09639*.

Zhang, X.; Lin, Y.; Miao, M.; Lou, J.; Li, J.; and Chen, X. 2025. Zeroth-Order Federated Private Tuning for Pretrained Large Language Models. In *Australasian Conference on Information Security and Privacy*, 285–306. Springer.

Zhao, P.; Zhang, H.; Yu, Q.; Wang, Z.; Geng, Y.; Fu, F.; Yang, L.; Zhang, W.; and Cui, B. 2024. Retrieval-augmented generation for ai-generated content: A survey. *arXiv preprint arXiv:2402.19473*.

Zhu, C.; Cheng, Y.; Gan, Z.; Sun, S.; Goldstein, T.; and Liu, J. 2019. Freelb: Enhanced adversarial training for natural language understanding. *arXiv preprint arXiv:1909.11764*.