Reconstruction Attack-Resistant Inference Paradigm for LLM Cloud Services

Zipeng Ye[1], Wenjian Luo[1*], Qi Zhou[1], Yubo Tang[1]

[1]Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies,
Institute of Cyberspace Security, School of Computer Science and Technology,
Harbin Institute of Technology, Shenzhen, China
22B351009@stu.hit.edu.cn, luowenjian@hit.edu.cn, {22S051036, 22S151061}@stu.hit.edu.cn

Abstract

Large language models (LLMs) have seen remarkable growth in recent years. To leverage convenient LLM cloud services, users are inevitably to upload their prompts. Additionally, for tasks such as translation, reading comprehension, and summarization, associated files or context are inherently needed, whether or not they contain user privacy information. Despite the rapid progress in LLM capabilities, research on preserving user privacy during inference has been relatively scarce. To this end, this paper conducts some exploratory research in this domain. Firstly, we show that (1) the embedding space of tokens is highly sparse, and (2) LLMs primarily function in the orthogonal subspace of embedding space, these two factors making privacy extremely vulnerable. Then, we analyze the structural characteristics of LLMs and design a distributed privacy-preserving inference paradigm which can effectively resist privacy attacks. Finally, we perform a thorough evaluation of the defended models on mainstream tasks and find that low-bit quantization techniques can be effectively combined with our inference paradigm, achieving a balance between privacy, utility, and runtime memory efficiency.

Introduction

In recent years, LLMs have made significant strides, allowing machines to perform various tasks using natural language instructions (Radford et al. 2019; Touvron et al. 2023). Despite the simple chatting uses, existing work has shown that supplying some extra prompts is beneficial for fully unleashing the potential of LLMs (e.g., in-context learning) (Brown et al. 2020). In particular, for some context-based tasks such as translation, reading comprehension and summary extraction, users inherently need to supply relevant information (e.g., by using RAG (Lewis et al. 2020)) from their personal databases as part of the prompt to the LLM APIs. A typical example is the integration of the latest GPTs (Achiam et al. 2023) in Microsoft Word and Excel, which are two widely used software across the globe. Users can simply select a portion of text or data and GPT can automatically treat them as contexts for various effective operations such as translation, continuation, or computation. This greatly enhances productivity. However, when text or data involves industry, business, or personal privacy—common in Office documents—using LLM cloud services as an auxiliary tool can pose privacy risks. Similar examples include using LLM-based coding tools and meeting summarization tools within organizations.

It appears that we are trapped in a dilemma: to benefit from the convenient cloud services of LLMs, we must compromise on privacy. A straightforward solution is to deploy LLMs on users' personal devices (Lin et al. 2024). However, not all LLM service providers are willing to do this. Further, users may also lack the hardware resources necessary to deploy and run LLMs locally. There is also another potentially viable method, i.e., differential privacy (DP) (Dwork 2006), which ensures privacy by carefully designed perturbations and has shown promise in several LLM training and fine-tuning tasks (Li, Tan, and Liu 2023; Liu et al. 2024). However, Hu et al. (Hu et al. 2024) argue that even a privacy budget in DP that was originally sufficient for protecting privacy can lead to complete privacy leakage when adversaries enhance the attacks, thus rendering the original privacy guarantees limiting.

In the inference phase, perturbation-based methods typically mitigate the leakage of privacy by perturbing or replacing the token embeddings (Zhang et al. 2024; Edemacu and Wu 2024). Nevertheless, we hold a slightly negative outlook towards the direct use of these methods in LLMs' inference phase. In this paper, through a comprehensive analysis, we will demonstrate that only substantial perturbations can effectively prevent adversaries from recovering the original data, while such perturbations can lead to a significant decline in model utility on challenging tasks (e.g., math, and we believe there are scenarios where users upload files or data and let the LLMs perform some statistics or calculations on the information contained within). In our perspective, a practical privacy-preserving method should meet the following criteria: (1) it is effective in resisting advanced attacks; (2) it minimally impacts the utility of LLMs; (3) it is easy to implement. Through an in-depth analysis of the structural characteristics of mainstream open-source LLMs, this paper proposes a novel method that simultaneously fulfills these three requirements to a certain extent.

Our Contribution. We propose a privacy-preserving inference paradigm for LLM cloud services and test its perfor-

*Corresponding author

mance across various tasks including general benchmarks, common-sense reasoning, mathematics, coding, and reading comprehension, with few-shot (Brown et al. 2020), zero-shot or chain-of-thought (CoT) (Wei et al. 2022) settings. Our contributions can be summarized as follows:

- We empirically show that the token embedding space is highly sparse, with embeddings of different tokens maintaining a significant distance from each other. Additionally, LLMs rarely modify the projection of hidden states in shallow layers. These two factors are the primary causes for the difficulty in safeguarding user privacy, also for this reason, we demonstrate that simply perturbing the embeddings is insufficient to effectively defend against reconstruction attacks.
- In conjunction with our analysis on the model structure, we propose a distributed privacy-preserving inference paradigm. Our method enhances the difficulty of attacks by employing a direction-maintained stochastic scaling transformation of the hidden states along with an adaptive compensation mechanism, thereby ensuring privacy without compromising utility.
- We validate the effectiveness and practicality of the proposed method through extensive experiments. Additionally, we find that the proposed defense method exhibits strong compatibility with low-bit quantization techniques, without necessitating any post-quantization calibrations. Our quantized defense strategy can further provide a balanced guarantee for privacy, model utility, and memory efficiency.

Methodology

Threat Model

For the threat model, we assume the victims are users of LLM cloud services who want to obtain the desired feedback by accessing the provided APIs with prompts. Concurrently, we consider the adversary to be a potentially malicious service provider. The adversary aims to obtain users' original data through carefully designed attack strategies when privacy-preserving methods are adopted by the users. Since the most commonly employed defense mechanism currently involves randomly perturbing the token embeddings or hidden states (Edemacu and Wu 2024), we assume that adversaries are capable of adopting advanced attack strategies against perturbation-based defense mechanisms. The overview of the threat model is shown in Fig. 1 (a).

In Fig. 1 (a), users incorporate some text from personal database into the prompt as context (e.g., obtained by RAG (Lewis et al. 2020)). Ideally, the LLM should infer from this context that the user is currently located in Hong Kong and proceed to design a route from Hong Kong to Taipei. Concurrently, some small, segmented modules are deployed at the user's end (Zhou et al. 2023; Mai et al. 2024), to protect user privacy through the application of random perturbations to either embeddings or hidden states. On the server side, an adversary, while interactively providing LLM services, employs advanced inversion attack methods to reconstruct user's original data (Qu et al. 2021). The green box in Fig. 1 (a) indicates scenarios where the adversary is unable to reconstruct the data, signifying that privacy is preserved; conversely, the red box denotes situations where privacy is compromised. Fig. 1 (b) shows the goal of the defense *(i.e., the goal of this paper): server can still provide the accurate responses while being unable to obtain the privacy even using advanced attacks.*

Empirical Study of Privacy Vulnerabilities in LLMs

In this part, we will illustrate through two interesting findings why it is challenging to effectively safeguard user data while maintaining the utility of LLMs, and without the in-depth analysis as well as the careful design, user privacy is quite vulnerable in cloud service scenarios.

Sparsity of Embedding Space Currently, the tokenizer of open-source LLMs, represented by Llama (Dubey et al. 2024), has a vocabulary size of more than 100,000 tokens, while Gemma (Team et al. 2024) boasts a vocabulary size of around 250,000 tokens. In the face of such a vast number of tokens, one might naturally inquire: *do the embeddings of these tokens cluster densely?* Contrary to this intuition, the embeddings of these tokens are, in fact, fairly sparsely distributed. In support of this, we design an experiment as follows. Considering the $(n-1)$-dimensional probability simplex whose vertices satisfy:

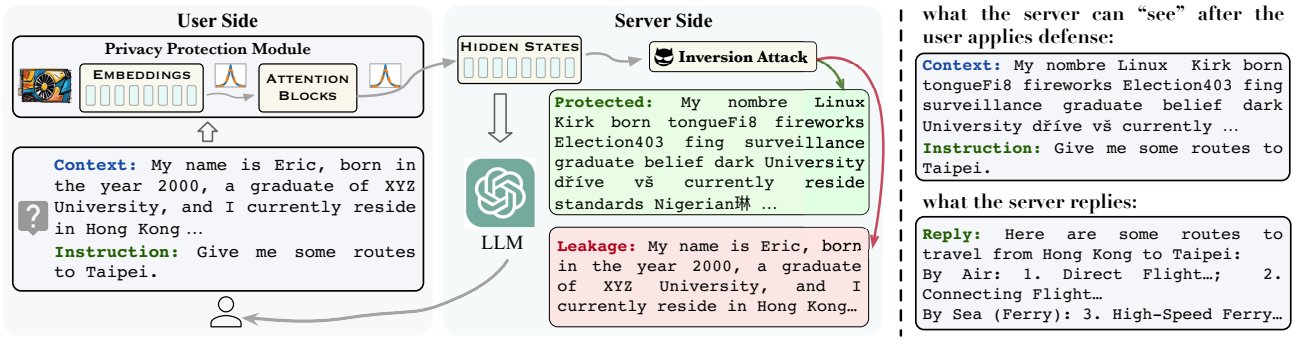
$$\left\{ w \in \mathbb{R}^n | \sum_{i=1}^{n} w_i = 1 \text{ and } w_i \geq 0 \text{ for } i = 1, \cdots, n \right\} \quad (1)$$

Obviously, if embedding space is very dense, when convex combinations with different weights w_i are applied to different embeddings E_i (where E_i is the embedding vector of i-th token), the resulting new vectors $\sum_{i=1}^{n} w_i E_i$ are more likely to approximate other embeddings, rather than consistently maintaining the closest proximity to $\{E_i\}_{i=1}^{n}$. In light of this perspective, we randomly select embeddings from n distinct tokens and subsequently sample weight w from the $(n-1)$ -simplex. For each vector $\sum_{i=1}^{n} w_i E_i$ resulting from the random convex combination of $\{E_i\}_{i=1}^{n}$, we identify the nearest token $\bar{T}$ (i.e., the embedding of $\bar{T}$ is closest to $\sum_{i=1}^{n} w_i E_i$) in the entire vocabulary list. By repeating this random process N times, we calculate the average Inclusion Ratio (IR) as follows:

$$\text{IR} = \frac{1}{N} \sum_{k=1}^{N} \mathbb{I}_{\Theta^{(k)}}(\bar{T}^{(k)}), \quad (2)$$

where $\Theta^{(k)}$ is the set with n tokens selected in the k-th round for the convex combination, and $\bar{T}^{(k)}$ is the identified nearest token in the k-th round. Indicator function $\mathbb{I}(\cdot)$ returns 1 if $\bar{T}^{(k)} \in \Theta^{(k)}$ else 0.

We set $N = 10,000$ for each n, and test on four open-source LLMs: Mistral (Jiang et al. 2023), Llama-3 (Dubey et al. 2024), Gemma-2 (Team et al. 2024) and Phi-3 (Abdin et al. 2024). Results are shown in Fig. 2. When $n \leq 8$, for all randomly sampled weights for convex combination, the token closest to the resulting vector is almost included



(a) (b)

Figure 1: Overview of the threat model, where (a) users aim to obtain LLMs services while safeguarding their privacy, whereas adversaries seek to obtain user privacy during the provision of services; (b) shows the ideal scenario where the server can respond accurately without being able to see the data.

within set $\Theta^{(k)}$. Furthermore, except for Gemma, such a phenomenon persists for the other three models when n is increased to 32. We contend that these findings strongly demonstrate that the embedding space is sparse, as *a certain number of embeddings, combined convexly in any manner, do not approximate any other tokens except themselves.* This also implies a high degree of discriminability among the embeddings corresponding to distinct tokens.

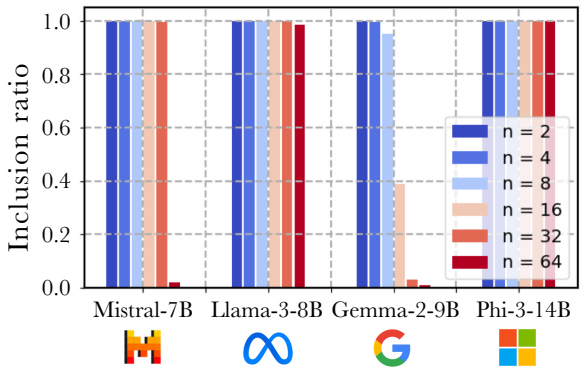


Figure 2: IR of resulting vector within the original token set, where each is statistically obtained on 10,000 experiments.

Shallow layers of LLMs change direction slightly in embedding space. We now adopt the perspective of an adversary to propose a practical attack method. In this context, we do not consider the plaintext scenario (where users directly transmit data as prompts) but rather the scenario where users only send the hidden states $\mathbf{h} \in \mathbb{R}^{l \times d}$ to the server, where l is the length of the tokenized prompt and d is the size of hidden vector. The hidden states are derived from several attention layers deployed on the user's end, i.e., $\mathbf{h} = F(\mathcal{E}) = f_m \circ \cdots \circ f_2 \circ f_1(\mathcal{E})$, where $\mathcal{E}$ is the ordered set of token embeddings from user prompt and f_i represents the i-th layer (Vaswani et al. 2017) in LLM. Then the optimization objective of the adversary can be expressed similarly to (Li, Tan, and Liu 2023):

$$\mathcal{E}^* = \arg\min_{\mathcal{E}'} \mathcal{L}\left(F(\mathcal{E}'), F(\mathcal{E})\right), \tag{3}$$

where $\mathcal{L}(\cdot)$ measures the distance between the reconstructed hidden states $\mathbf{h}'$ and the ground truth $\mathbf{h}$. Conventionally, we utilize gradient descent to update the dummy $\mathcal{E}'$ by minimizing the distance specified in (3), thereby obtaining the optimal $\mathcal{E}^*$. Subsequently, we apply the cosine matching, as previously introduced, to reconstruct tokens by the optimized $\mathcal{E}^*$. While we will later discuss the performance of this attack, we first pose an intriguing question: *What results might we obtain if we hypothesize $\mathcal{E}^* = \mathbf{h}$, followed by the direct application of cosine matching?* That is, we hypothesize that the user transmits hidden states $\mathbf{h}$, processed through m attention blocks, to the server, while an adversary directly assumes $\mathcal{E}^* = \mathbf{h}$ and performs cosine matching to obtain l tokens with the nearest directions to $\mathbf{h}$. We present results for Llama in Table 1 (column "w/o"), deferring deeper analysis to the next section, which informs the development of our defense methods, and additional results for other models can be found in the Appendix D.1.

We use Rouge (Lin 2004) to assess the similarity between reconstructed and original texts. As shown in Table 1, even without any updates to $\mathcal{E}'$, the adversary can obtain nearly all private information by user's hidden states $\mathbf{h}$ which is mapped through 10 attention blocks (blue text in Table 1). Such a result strongly suggests that the shallow layers of LLMs only minimally alter the direction in embedding space, thus making privacy susceptible to leakage. Moreover, when the adversary choose to optimize $\mathcal{E}'$ by gradient descent, even after passing through more layers, the essence of the original text is almost entirely reconstructed (see the last row in Table 1), which significantly underscores the vulnerability of privacy. The details about the attack implementation can be found in Appendix C.1. (We would like to clarify that part of a similar concept mentioned above appears in another concurrent submission of ours, which is still under review and not yet publicly available. For the sake of completeness and academic integrity, we reiterate the relevant concept in this paper and place the associated results in

	$m=1$		$m=5$		$m=10$		$m=15$		$m=20$		$m=25$	
	w/o	opt	w/o	opt	w/o	opt	w/o	opt	w/o	opt	w/o	opt
Rouge-1	1.00	1.00	0.96	1.00	0.88	0.91	0.67	0.93	0.40	0.84	0.23	0.84
Rouge-2	1.00	1.00	0.93	1.00	0.73	0.84	0.50	0.82	0.25	0.69	0.04	0.69
Rouge-L	1.00	1.00	0.96	1.00	0.88	0.91	0.67	0.93	0.40	0.84	0.23	0.84

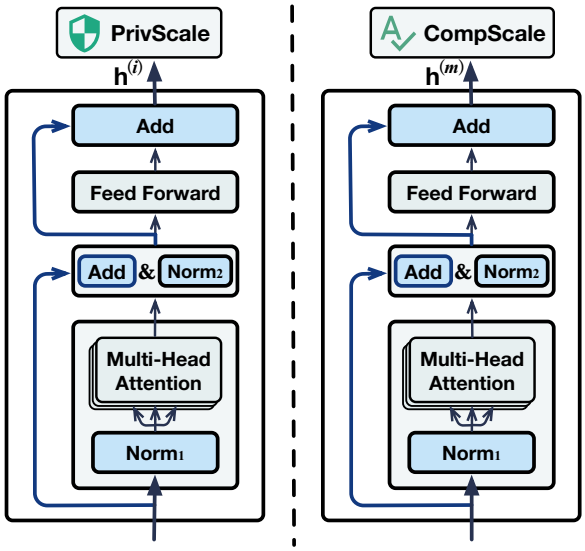
Truth	Apple Inc is an American multinational corporation and technology company headquartered in Cupertino, California, in Silicon Valley. It is best known for its consumer electronics, software, and services.
m=10, w/o	Apple Inc is an American multinational corporation and technology company headquartered in Cupertinouk Californiashaw in Silicon Valley. It is best knownCA its consumer electronics, software,x and services.
m=10, opt	Apple Inc is an American multinational corporation and technology company headquartered in CupertinoGray Californiavirt in Silicon Valley. It is best known for its consumer electronics gating software0 and services.
m=25, w/o	Apple battalionstatesAn American Milton_testing bezTechnology companygrad_levelsDemon\Web plaza NOT vitamin Silicon,valueDean He reass best known tx consumerelectronics Gong software,$ produk,$Dean
m=25, opt	Apple Inc is an American multinational companies AND technology companies headquartered in Cupertino' California/ IN Silicon Valley. It is best known for its consumer electronics—for softwareTechnology and Services.

Table 1: Quantitative and qualitative results of attacks on Llama-3-8B with (column "opt") or without (column "w/o") gradient-based optimization as user employs m attention layers.

the Appendix to minimize content overlap).

Privacy Enhancement and Utility Compensation

In this section, we will first elucidate why the hidden states processed through multiple attention blocks can still directly leak privacy. Based on this understanding, we will design privacy-enhancing method to effectively resist adversarial reconstruction attacks.



(a) i-th layer, $i < m$ (b) m-th layer

Figure 3: Architecture inside a transformer, where (a) PrivScale module is adpoted by user in the first $m-1$ layers and (b) CompScale module is adpoted in the m-th layer.

Nowadays, mainstream decoder-based LLMs share a similar backbone. The architecture of transformer with residual blocks allows the model to break traditional constraints on the number of layers in neural networks, with the former providing scalability and the skip connections in the residual blocks enabling the training of very deep networks. The function of layer i in decoder-based LLMs (refer to Fig. 3) can be mathematically expressed as follows (Vaswani et al. 2017). Note that we do not make a strict distinction between MHA and other attention mechanisms (e.g., GQA) here.

$$\begin{cases} \mathbf{h}^{-} = \mathbf{h}^{(i-1)} + \underbrace{\mathrm{MHA}\left(\mathrm{RMSNorm}_1(\mathbf{h}^{(i-1)})\right)}_{\mathcal{J}_1}, \\ \mathbf{h}^{(i)} = \mathbf{h}^{-} + \underbrace{\mathrm{FFN}\left(\mathrm{RMSNorm}_2\left(\mathbf{h}^{-}\right)\right)}_{\mathcal{J}_2}, \end{cases} \tag{4}$$

where $\mathrm{MHA}(\cdot)$ function as the multi-head attention block and $\mathrm{FFN}(\cdot)$ function as the feed forward network. RMSNorm (Zhang and Sennrich 2019) is adopted in nearly all mainstream LLMs due to its computational efficiency, which satisfies $\mathrm{RMSNorm}(\mathbf{a}) = \mathbf{g} \odot \frac{\mathbf{a}}{\mathrm{RMS}(\mathbf{a})}$, where $\mathbf{g}$ is the scaling parameters. We now make the conjectures to elucidate the circumstances under which the forward propagation of hidden states would significantly leak privacy: In the shallow layers, the cumulative sum of $\mathcal{J}_1 + \mathcal{J}_2$ is always located near the orthogonal subspace of token's embedding space.

Appendix B.2 provides the evidences and analysis for the proposition, which reveals the underlying causes for the privacy vulnerabilities observed in different LLMs. Obviously, with the above proposition, even after forward propagation across several layers, the projections of hidden states in embedding space will barely be altered, leading to the direct leakage of privacy from the inner product-based cosine matching

In the field of distributed learning, Ye et al. (Ye et al. 2024) highlight from an optimization perspective that increasing the non-linearity of the model architecture will enhance the difficulty of privacy attacks. However, given the intricate nature of training LLMs, it is not feasible to redesign the model architecture and retrain from scratch. Consequently, the satisfactory defense must be plug-and-play. To achieve

this requirement and effectively resist attacks, we propose to increase the proportion of $\mathcal{J}_1$ or $\mathcal{J}_2$ in Eq. (4), thereby amplifying the function of the nonlinear modules. However, adjusting $\mathcal{J}_1$ or $\mathcal{J}_2$ without careful consideration would undoubtedly severely impact the model's usability. Hence, we have designed a novel method which realizes the aforementioned objectives by shrinking each hidden state in $\mathbf{h}^{(i-1)}$ (i.e., $\mathbf{h}_j^{(i-1)} \in \mathbb{R}^d$, $j = 1, \cdots, l$) in a direction-preserving manner. This method offers two main benefits: first, after shrinking $\mathbf{h}_j^{(i-1)}$, the internal RMSNorm_1 of the MHA will restore it to its original scale, minimizing the impact on MHA's functionality; second, the shrinking of $\mathbf{h}^{(i-1)}$ will not alter the magnitudes of $\mathcal{J}_1$ and $\mathcal{J}_2$ significantly thanks to the normalization modules, thus leading to $\mathbf{h}^-$ and $\mathbf{h}^{(i)}$ being more dominated by the non-linear structures. The theoretical analysis is provided in Appendix B.1, where it is demonstrated that our method causes the adversary's optimization objective less convex, making attacks harder to successfully implement.

Specifically, we apply a random scaling to the output of the first i-th layers (i.e., input of the $(i+1)$-th layer where $i < m$). Finally, we compensate for the shrinking by applying a direction-preserving amplification to the output of the m-th layer. Extensive experimental results will demonstrate that this form of direction-preserving scaling is effective in resisting attacks while guaranteeing usability of LLMs, including on several difficult tasks. The mathematical expression of our defense is given in the follows, where the output $\mathbf{h}^{(i)} \in \mathbb{R}^{l \times d}$ of i-th layer in Eq. 4 is re-expressed as:

$$\begin{cases} \mathbf{h}^{(i)} = (\mathbf{p}^{-1} \cdot \mathbf{1}_d^T) \odot \left[\mathbf{h}^- + \text{FFN}\left(\text{RN}_2\left(\mathbf{h}^-\right)\right)\right], & \text{if } i < m \\ \mathbf{h}^{(i)} = (\mathbf{c} \cdot \mathbf{1}_d^T) \odot \left[\mathbf{h}^- + \text{FFN}\left(\text{RN}_2\left(\mathbf{h}^-\right)\right)\right], & \text{if } i = m \end{cases} \tag{5}$$

where RN_2 refers to RMSNorm_2, and each entry in $\mathbf{p} \in \mathbb{R}^l$ is randomly sampled from $p_j \sim U[1, 1+\delta]$ for each token in a context of length l. The constant vector $\mathbf{c} = c \cdot \mathbf{1}_l$ has a compensation scalar c. In our experiments, c is determined by selecting the first 20 training samples from the GSM8K math task (with CoT), feeding them into the privacy-enhanced inference model, and then performing a simple search for c within a given range to maximize accuracy on these 20 questions. This procedure is quick, usually completing within a few minutes.

Overall, in our distributed inference paradigm designed to resist reconstruction attacks, a total of m layers of privacy-enhancing and utility-compensating modules are deployed at the user-side. Further, in next section, we will show that low-bit quantization can be directly applied to these m layers, without necessitating post-quantization calibrations.

Experiments

Implementation Settings

Models, Tasks and Metrics. We use five instructed models to evaluate our method, including Mistral-7B-v0.3, Llama-3-8B, Gemma-2-9B, Phi-3-14B and Llama-3-70B-AWQ, and use six tasks for different privacy-preserving evaluations. Specifically, we protect all context for HellaSwag (Zellers et al. 2019), BoolQ (Clark et al. 2019), GSM8K (Cobbe et al. 2021) and HumanEval (Chen et al. 2021). In addition, we protect few-shot examples like (Tang et al. 2024) for tasks which employ few-shot learning, including MMLU (Hendrycks et al. 2021) and BBH (Suzgun et al. 2022). In Appendix C.3, we present a clear depiction of the protected part in these tasks and encourage readers to review. For evaluating the attack (with optimization), we use Rouge-1, Rouge-2 and Rouge-L (Lin 2004), where Rouge-1 measures the word-level (1-gram) reconstruction capability while Rouge-2 measures phrase-level (2-gram) and Rouge-L measures Longest Common Subsequence (LCS).

Criteria for Parameter Selection. We investigate the influence of δ for $\mathbf{p}$ in (5) on the quality of the reconstructions (we can search for the appropriate δ through conducting attack and defense locally by the m local layers). We use contexts in typical reading comprehension task (BoolQ) as targets and the statistical results are shown in Fig. 4 (a). Fig. 4 (b) proves that with the conditions of Rouge-1 < 0.5, Rouge-2 < 0.3, Rouge-L < 0.5, it is sufficient for the reconstruction to compromise a significant amount of information from the original data (more results are in Appendix D.4). According to this, as well as the results in Fig. 4 (a), we set δ to $[0.30, 0.20, 0.35, 0.50, 0.425]$ for Mistral-7B-v0.3, Llama-3-8B, Gemma-2-9B, Phi-3-14B and Llama-3-70B-AWQ, respectively.

Taking into account the requirement to counteract an adversary's random guessing, as well as the computational capabilities of user devices, we have configured the number of local layers $m = 10$. With a total of 9 (i.e., $m - 1$) consecutive layers, each accompanied by a distinct random scaling transformation applied to the hidden states corresponding to every token (and re-randomized for each inference), we believe this setup is sufficient to prevent an adversary from accurately guessing the specific scaling magnitude applied to the victim's data. As for the compensation scalar c, the results of rough search are shown in Fig. 4 (c), and based on this, the employed c is $[1.5, 1.0, 1.5, 2.0, 2.0]$, respectively. More about the experimental setup of Fig. 4 is given in the Appendix C.2. And in the future, we will delve into the investigation of more refined strategies for noise insertion based on the degree of module non-linearity, as well as explore configurations with smaller m.

Resisting Attacks

In this part, we assess the proposed method on resisting reconstruction attacks. The quantitative results are presented in Table 2, and the qualitative results are given in Appendix D.2. More experimental results are given in Appendix D.3 and D.4, including the attack results without countermeasure, as well as resisting attacks across various contexts from different datasets.

In Table 2, all Rouge scores meet the criteria outlined in the previous part. Furthermore, as indicated in Fig. 7 (Appendix D.2), our proposed defense method significantly safeguards a substantial amount of private information for all LLMs, even in cases (e.g., Llama-3-70B and Phi-3-14B) where, the Rouge-1 and Rouge-L scores of these reconstructions slightly over 0.5. These results substantiate the efficacy

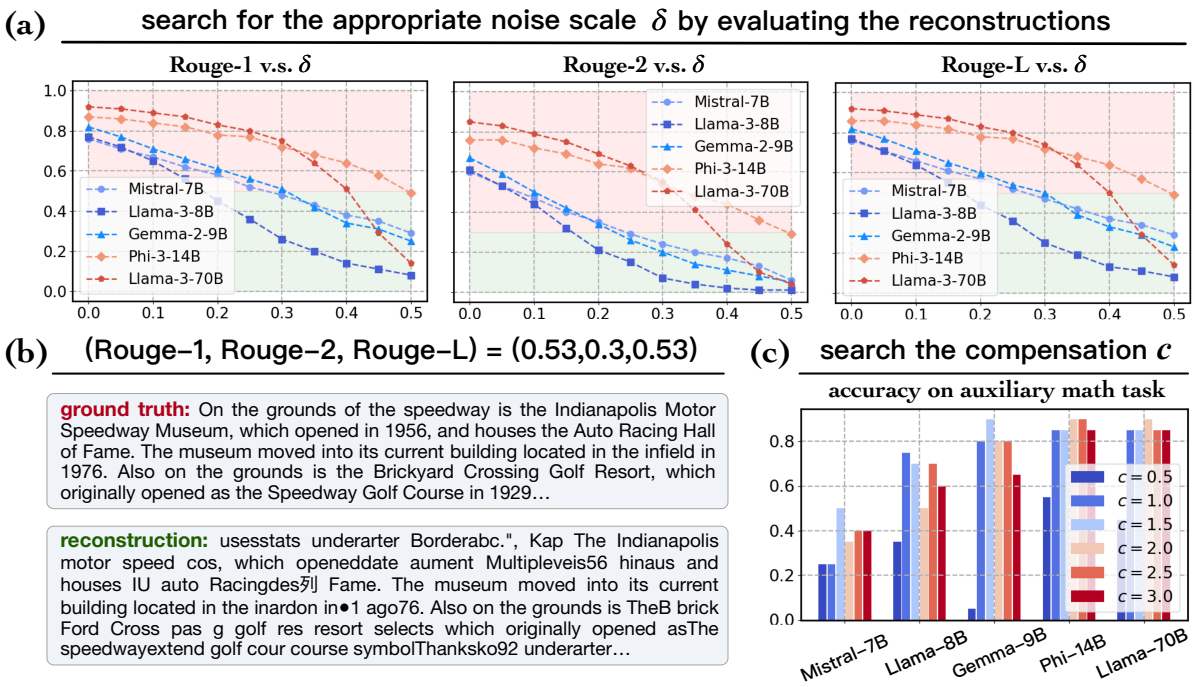


Figure 4: Algorithm parameters selection, where (a) illustrates the Rouge scores with different noise scale δ, with Rouge-1 < 0.5, Rouge-2 < 0.3, Rouge-L < 0.5 considered as privacy thresholds in this paper; (b) presents an attack result with (Rouge-1, Rouge-2, Rouge-L)=(0.53,0.3,0.53); (c) shows the accuracies on math task (first 20 training data of GSM8K) with different compensation c.

of our method in resisting attacks.

	Rouge-1	Rouge-2	Rouge-L
Mistral-7B	0.48	0.24	0.47
Llama-3-8B	0.45	0.21	0.44
Gemma-2-9B	0.42	0.14	0.39
Phi-3-14B	0.49	0.29	0.49
Llama-3-70B	0.39	0.17	0.39

Table 2: Rouge scores when using defense

Impact on Utility

We now need to consider whether a LLM can still function effectively after the "protection" of critical information, particularly in tasks involving math or code where content such as numbers and variables are decisive for the answers. Consequently, we have to deeply evaluate the remaining performance of models equipped with the proposed defense mechanism across various tasks. Simultaneously, we investigate the impact on model performance of directly perturbing embeddings or replacing tokens by nearest neighbors (see Appendix D.5 for details), with experimental results indicating that these strategies severely compromise performance, especially in coding and mathematical tasks, even when the perturbation scale is insufficient to counter attacks.

Choice-based tasks. Choice-based tasks involve choosing the correct answer from multiple choices (here we consider BoolQ (Clark et al. 2019) as a choice-based task, despite its responding with True or False rather than an explicit choice). In HellaSwag (commonsense reasoning, 0-shot) and BoolQ (reading comprehension, 0-shot), we apply privacy-preserving defenses to all context, which serves as the direct basis for the model's responses. In MMLU (57 subjects, 1-shot for Llama-70B and 5-shot for others), we treat all examples as privacy like (Tang et al. 2024) and protect them. Experimental results are presented in Table 3. For all experiments within the same task, we use the same prompts. Obviously, after applying defense, LLMs maintain quite good performance across these choice-based tasks. We also showcase the performance of LLMs across four subcategories of the MMLU. The results indicate that our method will not significantly degrade the performance of LLMs on a particular category.

Non choice-based tasks. In this part, we evaluate model's performance on the math task GSM8K (0-shot, with CoT) and the code task HumanEval (0-shot, pass@1). We apply protection directly to the context upon which all responses in GSM8K and HumanEval rely (see Appendix C.3). Results are presented in Table 4. Compared to choice-based tasks, there is a slightly greater performance decline in math and coding tasks, due to these tasks being more granular in nature and we have protected all their contexts. Even so, these LLMs remain effective, as that even after applying defense, their performance is either superior or comparable to that of slightly smaller models.

	HellaSwag		**BoolQ**		**MMLU**		◇ STEM		◇ Human		◇ Social		◇ Other	
	w/o	def	w/o	def	w/o	def	w/o	def	w/o	def	w/o	def	w/o	def
Mistral-7B	66.3	61.7	85.1	82.9	60.1	59.4	48.8	49.3	57.4	56.0	69.3	68.3	66.7	66.0
Llama-8B	66.7	65.4	84.3	83.0	65.8	65.2	55.8	54.6	60.9	60.5	76.0	75.5	73.3	72.9
Gemma-9B	81.9	80.1	89.2	87.7	72.2	72.1	65.7	65.0	66.1	67.2	83.5	82.7	76.8	76.5
Phi-14B	89.8	87.0	88.7	85.2	76.9	75.3	69.5	68.2	73.4	70.7	85.8	84.9	80.9	80.0
Llama-70B	85.1	83.0	89.7	83.2	77.7	74.0	71.6	70.5	72.8	67.3	86.6	82.2	82.4	79.8

Table 3: Accuracies of different tasks, where: "w/o" not using defense, "def" using defense.

	GSM8K (0-shot, CoT)						**HumanEval** (pass@1)						**BoolQ** (0-shot)			
	w/o	def	d-8	d-4	$L(\alpha)$	NR	w/o	def	d-8	d-4	$L(\alpha)$	NR	d-8	d-4	$L(\alpha)$	NR
Mistral-7B	54.8	46.8	46.6	43.4	2.1	3.5	38.4	34.1	39.0	38.4	5.5	3.0	82.6	82.8	42.6	73.1
Llama-8B	77.8	72.6	73.2	70.7	2.0	5.5	55.5	51.2	50.0	47.6	0	0	82.7	81.1	56.6	76.5
Gemma-9B	86.4	84.3	84.5	85.8	1.7	3.8	63.4	58.5	57.3	57.3	0	21.3	87.9	87.8	64.2	72.8
Phi-14B	91.1	85.2	85.2	78.1	2.3	4.5	70.1	64.6	63.4	58.5	4.9	4.3	84.7	82.8	70.5	76.5
Llama-70B	92.9	-	-	86.4	1.5	7.2	78.7	-	-	71.3	1.2	2.4	-	83.2	45.0	84.9

Table 4: Accuracies under different settings: "d-8" and "d-4" for defense with 8-bit and 4-bit quantization, "$L(\alpha)$" for perturbing embeddings following $\alpha = 0.5$ in Table 7 (Appendix B.3), "NR" for nearest replacing.

Impact on few-shot learning. BBH evaluates models using few-shot examples, and these examples are crucial as they determine how LLMs organize chain-of-thought and generate responses. Unlike prior experiments, in this part, we show that for tasks where the performance is better with 3-shot compared to 1-shot (not all tasks benefit from more examples), the addition of defense to all 3 examples still yields superior performance over 1-shot without defense. This experiment demonstrates that LLMs can still effectively learn from examples even with our defense.

To this end, we only evaluate on a subset of tasks from the BBH where 3-shot outperforms 1-shot (details are in Appendix D.6). Obviously, in Table 5, after applying defense, these LLMs still "learn" examples effectively and outperform those using 1-shot learning without defense. Owing to the lengthy computation time, we only evaluate the first 20 questions for each task in BBH for Llama-70B, and this setting does not affect the analysis.

	BIG-Bench Hard (CoT)		
	w/o(3-shot)	def(3-shot)	w/o(1-shot)
Mistral-7B	55.0	52.7	46.7
Llama-8B	68.2	67.5	57.4
Gemma-9B	77.8	75.6	71.2
Phi-14B	73.5	68.3	61.6
Llama-70B	77.7	72.3	63.2

Table 5: Accuracies on selected tasks in BBH.

Impact of quantization, perturbation and replacement. We select three representative tasks—math, coding and reading comprehension—to investigate the influence of applying low-bit quantization to the user-side modules when using our defense (see Table 4, note that the Llama-70B we used is downloaded from Hugging Face (Wolf et al. 2020) and is already quantized to 4-bit by AWQ). We also evaluate the impact on model utility by introducing perturbations to the embeddings, as well as replacing each token with its nearest token in embedding space (column "NR" in Table 4).

In Table 4, using our defense with 8-bit quantization only minimally affects model performance. However, 4-bit quantization can lead to a noticeable decline on certain tasks (in red). In contrast, perturbing embeddings with $\alpha = 0.5$ (Table 7 in Appendix B.3), which almost completely fails to protect privacy, yet significantly degrades usability, especially for math and coding tasks. As for the nearest replacing, a similar result is observed, presumably because the performance of math and coding tasks depends on token-level granularity, whereas replacing tokens with the nearest neighbors has a relatively smaller influence on text comprehension (comparison before and after nearest replacing is in Appendix D.5).

We also report the runtime GPU memory usage under different quantization precisions (see Table 6). HQQ (Badri and Shaji 2023) is applied to all 10 local layers, except for Llama-70B-AWQ (already quantized by AWQ (Lin et al. 2024)). The GPU memory required for these 10 layers is shown in the middle of Table 6. Since embedding layer primarily involves memory access rather than dense floating-point computations, transferring it to device is optional.

In Table 6, even the 70B model requires a memory size which is affordable for mobile devices. With the advancement of on-device AI and the development of flagship AI chips (Tan and Cao 2021; Gerganov et al. 2023), we believe that the proof-of-concept proposed in this paper will help to achieve a balance between privacy, utility, and memory efficiency for the future of on-device AI.

Conclusion and Future Work

This paper exposes the significant vulnerability of user privacy when employing LLM cloud services, and we contend that the attack method employed herein can serve as a benchmark for related research. Meanwhile, to alleviate the privacy leakage, we introduce a plug-and-play distributed inference paradigm. Extensive experimental results have demon-

	FP/BF16	8-bit	4-bit	embed
Mistral-7B	4.06	2.03	1.02	0.25
Llama-8B	4.06	2.03	1.02	0.98
Gemma-9B	3.69	1.85	0.92	1.71
Phi-14B	6.35	3.17	1.59	0.31
Llama-70B	-	-	4.14	1.96

Table 6: Memory required by the user in GB, "embed" denotes the required memory for embedding layer.

strated that our method can effectively resist privacy attacks while maintaining the usability of the model.

However, our work has several limitations. Firstly, the coarse-grained nature of our privacy-preserving shrinking operation on hidden states could be improved. Actually, a more granular strategy could be designed based on the sequence length (hidden states closer to the end of the sequence are more impacted due to the cumulation of preceding hidden states) and the non-linearity of modules, which would further mitigate the compromise on model performance. Additionally, in a few scenarios, performance degration may occur after directly quantizing model to 4-bit, where post-quantization calibration might be helpful (Frantar et al. 2022). Moreover, our method requires local-server collaboration for inference, implying the local device must have some computational capability. We will focus on addressing these limitations in our future work.

Acknowledgements

This study is supported by the National Key R&D Program of China (Grant No. 2022YFB3102100), Shenzhen Fundamental Research Program (Grant No. JCYJ20220818102414030), Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies (Grant No. 2022B1212010005).

References

Abdin, M.; Jacobs, S. A.; Awan, A. A.; Aneja, J.; Awadallah, A.; Awadalla, H.; Bach, N.; Bahree, A.; Bakhtiari, A.; Behl, H.; et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Badri, H.; and Shaji, A. 2023. Half-Quadratic Quantization of Large Machine Learning Models.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, 1877–1901.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; de Oliveira Pinto, H. P.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; et al. 2021. Evaluating Large Language Models Trained on Code. *arXiv preprint arXiv:2107.03374*.

Clark, C.; Lee, K.; Chang, M.-W.; Kwiatkowski, T.; Collins, M.; and Toutanova, K. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dwork, C. 2006. Differential privacy. In *International Colloquium on Automata, Languages, and Programming*, 1–12. Springer.

Edemacu, K.; and Wu, X. 2024. Privacy preserving prompt engineering: A survey. *arXiv preprint arXiv:2404.06001*.

Frantar, E.; Ashkboos, S.; Hoefler, T.; and Alistarh, D. 2022. GPTQ: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*.

Gerganov, G.; et al. 2023. llama.cpp.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*.

Hu, J.; Du, J.; Wang, Z.; Pang, X.; Zhou, Y.; Sun, P.; and Ren, K. 2024. Does Differential Privacy Really Protect Federated Learning from Gradient Leakage Attacks? *IEEE Transactions on Mobile Computing*.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, 9459–9474.

Li, Y.; Tan, Z.; and Liu, Y. 2023. Privacy-preserving prompt tuning for large language model services. *arXiv preprint arXiv:2305.06212*.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, 74–81.

Lin, J.; Tang, J.; Tang, H.; Yang, S.; Chen, W.-M.; Wang, W.-C.; Xiao, G.; Dang, X.; Gan, C.; and Han, S. 2024. AWQ: Activation-aware Weight Quantization for On-Device LLM Compression and Acceleration. In *Proceedings of Machine Learning and Systems*, volume 6, 87–100.

Liu, Z.; Lou, J.; Bao, W.; Qin, Z.; and Ren, K. 2024. Differentially Private Zeroth-Order Methods for Scalable Large Language Model Finetuning. *arXiv preprint arXiv:2402.07818*.

Mai, P.; Yan, R.; Huang, Z.; Yang, Y.; and Pang, Y. 2024. Split-and-Denoise: Protect large language model inference with local differential privacy. In *International Conference on Machine Learning*.

Qu, C.; Kong, W.; Yang, L.; Zhang, M.; Bendersky, M.; and Najork, M. 2021. Natural language understanding with privacy-preserving bert. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 1488–1497.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Suzgun, M.; Scales, N.; Schärli, N.; Gehrmann, S.; Tay, Y.; Chung, H. W.; Chowdhery, A.; Le, Q. V.; Chi, E. H.; Zhou, D.; ; and Wei, J. 2022. Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them. *arXiv preprint arXiv:2210.09261*.

Tan, T.; and Cao, G. 2021. Efficient execution of deep neural networks on mobile devices with npu. In *Proceedings of the International Conference on Information Processing in Sensor Networks*, 283–298.

Tang, X.; Shin, R.; Inan, H. A.; Manoel, A.; Mireshghallah, F.; Lin, Z.; Gopi, S.; Kulkarni, J.; and Sim, R. 2024. Privacy-Preserving In-Context Learning with Differentially Private Few-Shot Generation. In *International Conference on Learning Representations*.

Team, G.; Mesnard, T.; Hardin, C.; Dadashi, R.; Bhupatiraju, S.; Pathak, S.; Sifre, L.; Rivière, M.; Kale, M. S.; Love, J.; et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

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*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention Is All You Need. In *Advances in Neural Information Processing Systems*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 24824–24837.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; and Others. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 38–45.

Ye, Z.; Luo, W.; Zhou, Q.; Zhu, Z.; Shi, Y.; and Jia, Y. 2024. Gradient Inversion Attacks: Impact Factors Analyses and Privacy Enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zellers, R.; Holtzman, A.; Bisk, Y.; Farhadi, A.; and Choi, Y. 2019. HellaSwag: Can a Machine Really Finish Your Sentence? In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Zhang, B.; and Sennrich, R. 2019. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, volume 32.

Zhang, X.; Fei, Y.; Kang, Y.; Chen, W.; Fan, L.; Jin, H.; and Yang, Q. 2024. No Free Lunch Theorem for Privacy-Preserving LLM Inference. *arXiv preprint arXiv:2405.20681*.

Zhou, X.; Lu, Y.; Ma, R.; Gui, T.; Wang, Y.; Ding, Y.; Zhang, Y.; Zhang, Q.; and Huang, X.-J. 2023. Textobfuscator: Making pre-trained language model a privacy protector via obfuscating word representations. In *Findings of the Association for Computational Linguistics: ACL 2023*, 5459–5473.