DP-GTR: Differentially Private Prompt Protection via Group Text Rewriting

Mingchen Li[1,♠], **Heng Fan**[1,♢], **Song Fu**[1,♢], **Junhua Ding**[2,♢], **Yunhe Feng**[1,♢]
University of North Texas
[1] Department of Computer Science and Engineering [2] Department of Data Science
♠ MingchenLi@my.unt.edu
♢ {heng.fan, song.fu, junhua.ding, yunhe.feng}@unt.edu

Abstract

Prompt privacy is crucial, especially when using online large language models (LLMs), due to the sensitive information often contained within prompts. While LLMs can enhance prompt privacy through text rewriting, existing methods primarily focus on document-level rewriting, neglecting the rich, multi-granular representations of text. This limitation restricts LLM utilization to specific tasks, overlooking their generalization and in-context learning capabilities, thus hindering practical application. To address this gap, we introduce DP-GTR, a novel three-stage framework that leverages local differential privacy (DP) and the composition theorem via *group text rewriting*. DP-GTR is the first framework to integrate both document-level and word-level information while exploiting in-context learning to simultaneously improve privacy and utility, effectively bridging local and global DP mechanisms at the individual data point level. Experiments on CommonSense QA and DocVQA demonstrate that DP-GTR outperforms existing approaches, achieving a superior privacy-utility trade-off. Furthermore, our framework is compatible with existing rewriting techniques, serving as a plug-in to enhance privacy protection. Our code is publicly available at github.com/ResponsibleAILab/DP-GTR.

1 Introduction

The rise of LLMs in natural language processing has catalyzed an urgent research focus on their security and privacy (Yao et al., 2024; Peng et al., 2025; Luo et al., 2025), including investigations into differential privacy (DP) techniques to mitigate the risk of sensitive information leakage (Edemacu and Wu, 2025; Abadi et al., 2016; Wu et al., 2023; Tang et al., 2023). While DP, the gold standard for computational privacy, has seen broad adoption in machine learning, existing text-based DP methods face significant challenges. These methods generally fall into four categories: training-based optimizations (e.g., DP-SGD (Abadi et al., 2016)), embedding perturbations (Feyisetan et al., 2020), document-level paraphrasing (Mattern et al., 2022), and in-context learning (ICL) enhancements (Wu et al., 2023). However, training-based approaches are computationally expensive, embedding perturbations can compromise semantic coherence, and ICL often neglects client-side prompt privacy.

Document paraphrasing offers a promising balance between privacy and utility. State-of-the-art methods achieve differentially private next-token generation using the exponential mechanism (EM) (McSherry and Talwar, 2007; Carvalho et al., 2023), replacing the standard softmax. Initial work employed decoder-only models like fine-tuned GPT-2 (Mattern et al., 2022), progressing to encoder-decoder (Igamberdiev and Habernal, 2023) and encoder-only architectures (e.g., BART, RoBERTa) (Meisenbacher et al., 2024b). DP-Prompt (Utpala et al., 2023) leverages prompt learning for zero-shot paraphrasing, and recent advancements combine DP post-processing with adversarial fine-tuning (Meisenbacher and Matthes, 2024). A critical limitation, however, persists: lack of fine-grained control over the privacy-utility trade-off.

Current EM-based DP methods provide only coarse-grained control via the privacy budget, hindering practical deployment. Most approaches (except DP-Prompt) also necessitate resource-intensive fine-tuning. While document-level paraphrasing preserves more contextual information than embedding perturbations, it often overlooks word-level privacy vulnerabilities. These limitations highlight the need for a training-free, fine-grained privacy solution that fully leverages textual information, a capability well-suited to the ICL paradigm of LLMs.

Prior work on DP in ICL has predominantly focused on server-side, global DP implementations, often using a "sample-and-aggregate" ap-

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 1573–1585
November 4-9, 2025

proach (Nissim et al., 2007a) to privately partition and aggregate context databases (Wu et al., 2023; Tang et al., 2023). Client-side prompt privatization, in contrast, requires the stronger guarantees of local DP (LDP), protecting individual data points rather than entire datasets. This distinction creates a significant gap between global and local DP in ICL, motivating the need for approaches that bridge it.

Addressing these gaps, we propose DP-GTR, a three-stage, differentially private prompt protection framework built upon a novel *Group Text Rewriting (GTR)* mechanism (see Figure 1). DP-GTR is designed to provide fine-grained control over the privacy-utility trade-off while remaining compatible with existing paraphrasing techniques. In Stage-1, GTR generates multiple client-side paraphrases of an input prompt, forming a "rewriting group" that preserves rich contextual information and enables bag-of-words-like count analysis. Notably, GTR connects local and global DP principles on the client-side. Stage-2 uses these counts for fine-grained privacy-utility control, identifying potentially sensitive private consensus keywords – words appearing frequently across paraphrases despite DP-driven variations. We mitigate this risk by releasing a fixed number of these keywords or using a differentially private aggregator, and select the lowest-perplexity paraphrase to maximize output quality. Stage-3 suppresses the identified keywords, limiting privacy leakage, and uses the selected paraphrase as an ICL example to improve utility. In addition, we evaluate DP-GTR in a realistic question-answering (QA) scenario, simulating real-world LLM usage.

Our key contributions are:

- We propose *Group Text Rewriting (GTR)*, a novel mechanism bridging local and global DP at the client-side prompt, enabling the integration of various DP techniques.
- We present DP-GTR, a three-stage prompt protection framework leveraging ICL for fine-grained privacy-utility control, compatible with existing paraphrasing methods.
- To our knowledge, we are the first to unify document-level and word-level privacy considerations within a single framework.
- We evaluate state-of-the-art DP paraphrasing methods in a realistic QA setting, demonstrating DP-GTR's superior privacy-utility trade-off compared to existing approaches.

2 Related Work

Global vs. Local DP: Differential privacy text sanitization methods are classified into Global Differential Privacy (Global-DP) and Local Differential Privacy (Local-DP) based on where the privacy mechanism is applied. In Global-DP, data is aggregated centrally before applying the privacy mechanism, while methods like DP-SGD use differentially private optimization techniques for training text models (Abadi et al., 2016; Ponomareva et al., 2022; Feyisetan et al., 2020). DP-ICL operates within a "sample-and-aggregate" framework by perturbing the embedding and vocabulary selected for release (Wu et al., 2023). In contrast, Local-DP incorporates the differential privacy mechanism before data reaches the centralized processor, typically affording stronger privacy protection (Duchi et al., 2013; Feyisetan et al., 2020).

Local-DP: Private document release methods are categorized into three tiers based on where noise is added: word-level, sentence-level, and document-level. At the word level, noise is added to word embeddings, and the perturbed vectors are then mapped to the nearest vocabulary word (Feyisetan et al., 2020; Xu et al., 2020; Yue et al., 2021). Carvalho et al. (2023) employ the exponential mechanism for token selection, while Chen et al. (2023a) propose customized token mappings for individual words. Moreover, Meisenbacher et al.'s (2024a) study generates multiple candidate perturbations using various word embedding models, and Mattern et al. (2022) highlights that word-level approaches inherently lack contextual information. Similarly, sentence-level methods inject noise into sentence embeddings (Reimers, 2019; Meehan et al., 2022).

Document-level: At the document level, paraphrasing technologies are grouped into three categories based on model architecture. Mattern et al. (2022) uses a decoder-only fine-tuned GPT-2 model. Later work adopts encoder-decoder models, such as a BART-based approach (Lewis, 2019) with sensitivity clipping via thresholding and pruning (Igamberdiev and Habernal, 2023). Encoder-only methods, like DP-MLM (Meisenbacher et al., 2024b), use a RoBERTa-based masked language model for fine-tuning. These methods require fine-tuning. In contrast, DP-Prompt (Utpala et al., 2023) introduces a zero-shot prompt learning paradigm using black-box LLMs, and Meisenbacher and Matthes (2024) employs post-processing and adversarial fine-tuning to enhance rewriting.

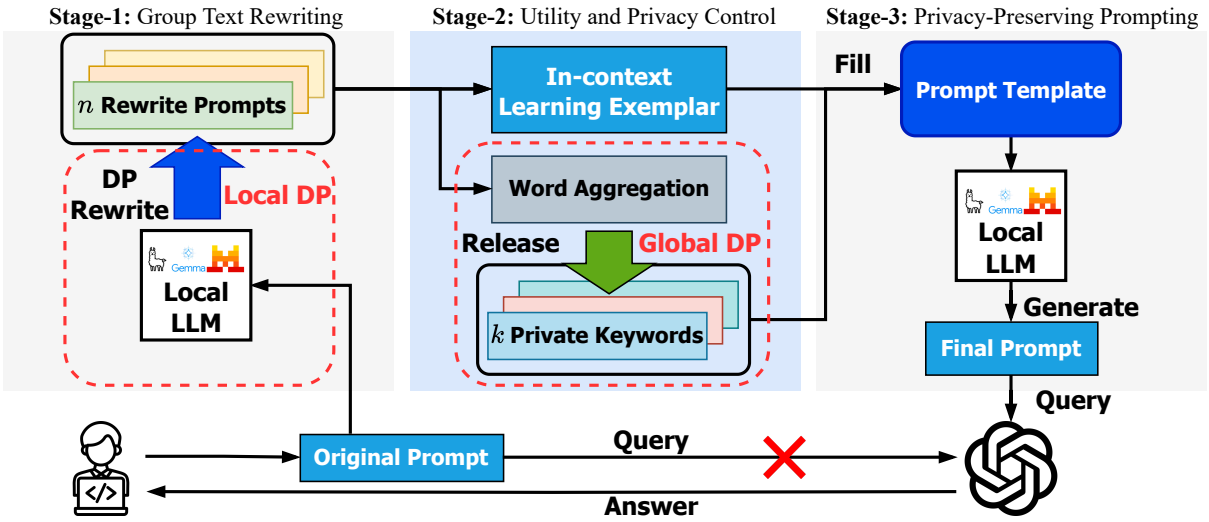


Figure 1: DP-GTR: A three-stage pipeline for Differentially Private prompt protection via Group Text Rewriting (GTR). Stage-1 generates n paraphrases of the original prompt using a DP paraphrasing mechanism. Stage-2 identifies the lowest-perplexity prompt as the ICL exemplar and aggregates word counts to release k private keywords. Stage-3 integrates these private keywords and the ICL exemplar into a prompt template for submission to the LLM, producing the final, differentially private prompt.

DP in ICL: The primary concern with applying DP in ICL is that LLMs are not inherently secure, potentially exposing sensitive context. Wu et al. (2023)'s DP-ICL perturbs embeddings and extracts keywords from subsampled datasets, while Tang et al. (2023) incorporates label information. Zheng et al. (2024) employ k-RR (Wang et al., 2017) to generate ICL answers, and Gao et al. (2024) aggregate next-token predictions from dataset shards. All the approaches follow a "sample-and-aggregate" framework (Nissim et al., 2007a), partitioning the data and applying private aggregation.

Our work, DP-GTR, draws on the principles of prompt learning and the "sample-and-aggregate" strategy from DP-Prompt and DP in ICL respectively. This one-shot in-context learning framework, analogous to global DP, obviates resource-intensive fine-tuning while enhancing both privacy protection and utility.

3 Preliminaries

Threat Model. We assume an adversary attacker on the side of the cloud-based model vendor, whose objective is to extract private information (such as personal identifiable information) from the confidential transmission content. The adversary's access is limited to a customized prompt supplied by the client, but they are free to facilitate their attack.

Pure Differential Privacy (DP) A randomized mechanism $\mathcal{M} : \mathcal{X} \to V$ satisfies ϵ-Pure DP if, for any neighboring datasets D and D' differing by at most one element, and any output $V \subseteq \text{Range}(\mathcal{M})$, the following inequality holds $\Pr[\mathcal{M}(D) = V] \leq e^{\epsilon} \cdot \Pr[\mathcal{M}(D') = V]$ (Dwork et al., 2006).

Local Differential Privacy Local DP applies a mechanism $\mathcal{M}$ to each individual data point $x, x' \in \mathcal{X}$ (where x and x' are considered neighboring in some sense), generating a local perturbation V before the data is submitted to the data center (Duchi et al., 2013; Dwork et al., 2006).

Metric Differential Privacy To improve the utility of DP, the indistinguishability of two outputs for x and x' can be scaled by the distance between their corresponding inputs (Alvim et al., 2018). A mechanism $\mathcal{M}$ satisfies ϵ-Metric DP if, for any inputs $x, x' \in \mathcal{X}$ and any output $V \subseteq \text{Range}(\mathcal{M})$, the following inequality holds:

$$\Pr[\mathcal{M}(x) = V] \leq e^{\epsilon \cdot d(x,x')} \cdot \Pr[\mathcal{M}(x') = V],$$

where $d(x, x')$ is a distance metric defined on $\mathcal{X}$.

Exponential Mechanism The *Exponential Mechanism* (EM) injects noise into scoring functions, making it suitable for non-numeric sensitive queries (McSherry and Talwar, 2007). Given a dataset D and a utility function $u : D \to V$, where V is the set of possible outputs, the mechanism $\mathcal{M}$ is defined as

$$\Pr[\mathcal{M}(D) = v] \propto \exp\left(\frac{\epsilon\, u(D, v)}{2\,\Delta u}\right),$$

where the sensitivity Δu is defined as

$$\Delta u = \max_{D,D',v} \big|u(D, v) - u(D', v)\big|,$$

and the maximum is taken over all neighboring datasets D and D' and all possible outputs $v \in V$.

Composition Property Differential privacy exhibits a robust *composition property*: when multiple DP mechanisms are applied sequentially to the same dataset, the overall privacy loss accumulates (Dwork et al., 2014). Let D be a dataset and let $M_1, M_2, \ldots, M_n$ be ϵ_i-DP mechanisms. The composed mechanism

$$M = M_n \circ M_{n-1} \circ \cdots \circ M_1$$

satisfies ϵ-DP with $\epsilon = \sum_{i=1}^{n} \epsilon_i$.

Post-Processing Property The *post-processing property* states that any function applied to the output of a DP mechanism preserves the same privacy guarantee (Dwork et al., 2014). If a mechanism $\mathcal{M} : \mathcal{X} \to V$ satisfies ϵ-DP, then for any function $F : V \to V'$, the composed mechanism $F \circ \mathcal{M}(D)$ also satisfies ϵ-DP.

DP-Guaranteed Paraphrasing Autoregressive language models (LMs) generate text *sequentially*, sampling tokens from a conditional likelihood distribution: $\prod_{i=1}^{n} \Pr[x_i \mid x_1, \ldots, x_{i-1}, C]$, where $C = (c_1, c_2, \ldots, c_m)$ is the context. At each step, a logit vector $u \in \mathbb{R}^{|\mathcal{V}|}$ is transformed into a probability distribution over the vocabulary $\mathcal{V}$ using a softmax function with temperature T:

$$p(v) = \frac{\exp(u_v/T)}{\sum_{w \in \mathcal{V}} \exp(u_w/T)}, \quad \forall\, v \in \mathcal{V}.$$

Prior work (Utpala et al., 2023; Mattern et al., 2022) has shown the equivalence between this softmax selection process and the Exponential Mechanism (EM) of differential privacy, where the utility function corresponds to the logits. Assuming LM is not pre-trained on the distribution of the data being protected, and that logits u_v are clipped to $[b_{\min}, b_{\max}]$, generating n tokens at temperature T provides a $(\frac{2n(b_{\max}-b_{\min})}{T})$-local DP (LDP) guarantee. This derives from the fact that each token selection, with a maximum logit difference of $(b_{\max} - b_{\min})$, incurs a privacy loss of $\frac{2(b_{\max}-b_{\min})}{T}$. Sequential composition over n tokens then yields the stated LDP bound for a single document paraphrase. Logit clipping and EM sampling ensure the generated sequence respects a well-defined pure LDP budget. See Appendix A for details.

4 DP-GTR

Existing document-level prompt sanitization methods often employ the EM for privacy-preserving rewriting. However, these coarse-grained approaches, relying on a single ϵ for the entire document (prompt) rewriting, struggle to balance privacy and utility. Critically, noise introduced during rewriting irreversibly alters textual elements, hindering utility recovery. Maintaining acceptable utility thus necessitates low initial noise levels, requiring a high privacy budget and consequently reducing actual privacy protection. Furthermore, a high privacy budget under the EM can even lead to complete data exposure. To address these limitations, we propose DP-GTR, a word- and document-level hybrid prompt privacy adopted framework that leverages group text rewriting and post-processing to enhance privacy while maintaining high utility under DP guarantees. DP-GTR enables low-noise paraphrasing to identify and suppress the generation of privacy-sensitive terms with high exposure.

4.1 DP-GTR Framework Overview

DP-GTR comprises three distinct stages, as illustrated in Figure 1. In Stage-1, a DP-guaranteed group text rewriting process explores diverse representations of the original prompt, generating n rewritten versions. Stage-2 leverages this group of rewritten prompts in a parallel process. First, it identifies the lowest-perplexity rewritten prompt as an in-context learning exemplar, effectively selecting the most confident paraphrase. Concurrently, it aggregates word counts across the rewritten prompts and releases k private keywords shared within the group. Finally, Stage-3 employs a prompt template. This template is populated with both the selected in-context learning exemplar and the released private keywords. The filled template is then fed to the LLM to generate the final prompt, effectively mitigating the risk of directly revealing sensitive information from the original prompt.

4.2 Stage-1: Group Text Rewriting

DP-GTR employs group text rewriting to achieve finer-grained control over privacy and utility compared to document-level methods. Effective prompt sanitization requires considering both document-level context for overall meaning and word-level information for protecting sensitive terms. While LLMs excel with contextual input, directly using the original prompt compromises privacy.

Group text rewriting addresses this by generating a local paraphrased text database, effectively mitigating the limitations of both document-level rewriting and the absence of suitable contextual information. This database, consisting of multiple rewritten versions of the prompt, serves several key purposes. First, it provides richer information than a single rewrite, capturing diverse facets of the original prompt. Second, aggregating word counts across the rewrites facilitates the identification of shared, potentially sensitive keywords. More importantly, the group enables more aggressive post-processing and additional DP mechanisms, which, while not reducing the formal ϵ-DP bound, further mitigate the risk of sensitive information disclosure by eliminating sensitive identifiers and limiting real-world exposure. Specifically, generating a paraphrased group $\mathcal{P}$ of m documents, each with n tokens, incurs an $(mn)\epsilon_1$-DP privacy budget.

4.3 Stage-2: Utility and Privacy Control

DP-GTR achieves fine-grained control over utility and privacy by leveraging the group of rewritten prompts. Utility is enhanced through in-context learning, while privacy is preserved via private keyword analysis.

4.3.1 One-shot in-context learning for utility

LLMs exhibit a strong capacity for in-context learning (Brown et al., 2020), effectively learning from provided examples. To ensure the LLM understands the desired output format and content, contextual information is crucial. Furthermore, LLMs often demonstrate a preference for learning from their own generated content. Therefore, we employ a one-shot in-context learning approach to maximize the utility of the rewritten prompt.

Specifically, rather than using the original prompt, we select the lowest-perplexity paraphrase, P_{low}, from the generated group $\mathcal{P}$ as the exemplar for guiding final prompt generation. This paraphrase, representing the most coherent and representative information within the group, serves as the most effective learning example. Critically, this in-context learning process leverages the post-processing property of differential privacy, incurring no additional privacy budget. Formally, this can be seen as the temperature approaching zero as the privacy budget approaches infinity: $T = \lim_{\epsilon \to \infty} \frac{2(b_{\max} - b_{\min})}{\epsilon} \to 0$.

4.3.2 Consensus-aware privacy protection

Protecting prompt privacy requires identifying privacy-sensitive keywords. Unlike previous PII detection methods that focus on isolated words or phrases (Chen et al., 2023b,a), DP-GTR considers the overall *composition* of sentences to comprehensively capture privacy leakage risks. Due to the paraphrasing tendencies of LLMs, key pieces of information within these compositional relationships often reappear across different paraphrased examples. We define *consensus words* as those that appear repeatedly across multiple paraphrased prompts generated in Stage-1. This repetition is treated as a privacy signal, as words appearing frequently despite paraphrasing attempts are likely either (a) crucial to the document's meaning and difficult to alter without significant utility loss, or (b) inherently tied to sensitive or identifiable information (e.g., names, locations) that existing LDP methods struggle to effectively anonymize.

Consensus Keyword Extraction The paraphrased group generated in Stage-1 can reproduce large fragments, potentially "leaking" sensitive information. Inspired by the bag-of-words approach, we count word frequency (c) across the paraphrased sentences, forming a set of frequency counts $\mathcal{S} = \{(w_1, c_1), (w_2, c_2), \ldots, (w_k, c_k)\}$. We then release a fixed number (K) of keywords, $\mathcal{K} = \{k_1, k_2, \ldots, k_K\}$, without manual intervention. This can be achieved either through post-processing or by employing the Joint Exponential Mechanism (Joint-EM) (Gillenwater et al., 2022) with privacy budget ϵ_2 under the sample-and-aggregate framework (Nissim et al., 2007b). The consensus keyword extraction algorithm is detailed in Algorithm 1.

Post-Processing Release Due to the post-processing property of DP, keywords $\mathcal{K}$ can be directly released, incurring no additional differential privacy (NDP) budget. This allows for diverse downstream analyses without further privacy cost, maintaining the total budget at $(mn)\epsilon_1$-LDP. This approach is designated ***DP-GTR-NDP***.

Joint-EM Release Joint-EM provides a privacy-preserving DP mechanism for simultaneously releasing the top-K keywords (Gillenwater et al., 2022), making it a suitable alternative. This approach has a total privacy budget of $((mn)\epsilon_1 + \epsilon_2)$-LDP and is designated ***DP-GTR-JEM***.

Algorithm 1 Top-K Private Keywords Extraction

Require: $\{P_1, P_2, \ldots, P_M\}$: paraphrased documents; K: output word count; `method` $\in$ {`post-processing`, `Joint-EM`}; privacy budget ϵ

Ensure: Top-K highest-frequency words

for $i \leftarrow 1$ to M **do**
  $S_i \leftarrow$ SEPARATEBYSPACE(P_i)
  $S_i \leftarrow$ REMOVESTOPWORDS(S_i)
  `private_keywords` $\leftarrow \{\}$
  for all $w \in S_i$ **do**
    `private_keywords`$[w]$ `+= 1`
  end for
  SORTDESCENDING(`private_keywords`)
end for
if `method` = `post-processing` **then**
  return TOPK(`private_keywords`, K)
else if `method` = `Joint-EM` **then**
  return JOINTEM-TOPK (`private_keywords`,ϵ, K)
end if

4.4 Stage-3: Privacy-Preserving Prompting

To maximize utility while preserving prompt privacy, DP-GTR constructs the final prompt in Stage-3. Leveraging LLM prompt learning, particularly the stronger learning aptitude exhibited with negative commands (Zhong et al., 2024; Wei et al., 2022), we utilize the extracted consensus keywords to effectively *prevent* the generation of private information. This approach offers both practical and gentle privacy protection. Practically, we directly instruct the LLM to avoid generating the identified private keywords, eliminating the need for further word, token, or document modification, thus streamlining the process. Gentle privacy is achieved by strategically engineering prompts to selectively suppress model output, rather than relying on simple filtering rules or context-agnostic direct replacement.

To maximize utility, we incorporate the lowest-perplexity rewritten prompt, P_{low}, selected in Stage-2, as a one-shot in-context learning example. Simultaneously, to ensure privacy, we instruct the LLM to avoid generating the private keywords, $w_1, w_2, \ldots, w_k$, released in Stage-2. Our extracted keywords contain richer combinatorial information and global context, enabling this more nuanced control compared to other methods. The resulting prompt template is shown below.

Privacy-Preserving Prompt Template

Refer to the following question to generate a new question: <P_{low}> Avoid using the following tokens: <$w_1, w_2, ..., w_k$>

5 Experiment

5.1 Limitations of Current Metrics

Prior work on prompt privacy preservation, primarily focused on author obfuscation (Utpala et al., 2023) using datasets like *Yelp* and *IMDb*, typically evaluates privacy via adversarial classifiers attempting to identify the original author and utility through binary sentiment classification using BERT-based models (Kenton and Toutanova, 2019). These evaluations, however, are often coarse-grained and fail to capture nuanced changes in meaning or style. For instance, a severely degraded paraphrase like "!!!!!" might be deemed to protect the author's identity and preserve the original positive sentiment of "At least for me, this movie is good!!," despite a significant loss of information, bordering on hallucination. Such "protection," arising from factors like high temperature settings, specific formatting, autoregressive generation, or model limitations, highlights the inadequacy of these existing evaluation metrics for assessing real-world applicability, as they fail to penalize extreme modifications that compromise the prompt's informational content.

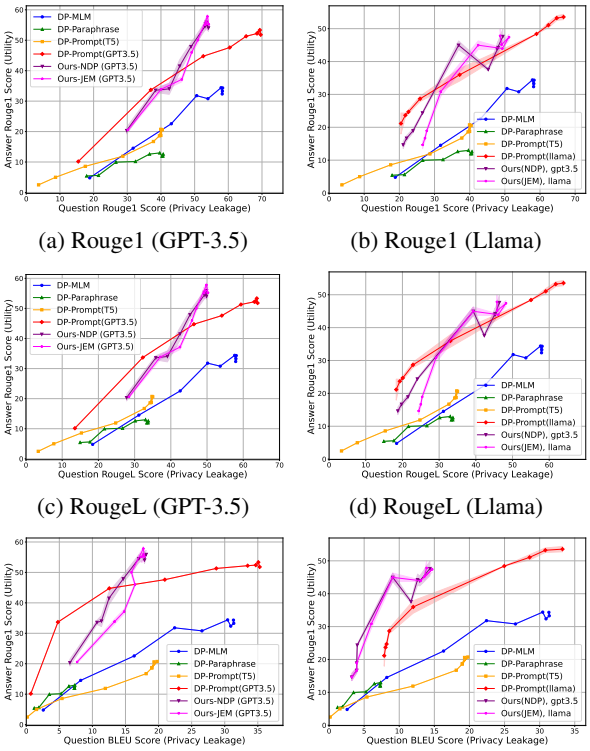
5.2 Experiment Setups

To evaluate prompt privacy and utility in a practical LLM service context, we propose an integrated question answering (QA) evaluation framework, conducting a single QA round to simultaneously measure both privacy and security. We use two QA datasets: the 5-choice closed-answer Commonsense QA (***CSQA***) (Talmor et al., 2019) and the open-answer PFL-DocVQA (***VQA***) (Tito et al., 2024), selecting 200 random items from each dataset's validation set. Note that VQA provides pre-extracted OCR tokens.

Integrated Evaluation. We simultaneously evaluate prompt privacy and utility. Privacy is measured by minimizing Rouge1, RougeL (Lin, 2004), and BLEU (Papineni et al., 2002) scores between the original prompt p and the sanitized prompt p', indicating ***greater privacy with more dissimilar prompts***. Utility is assessed using a GPT-3.5 (OpenAI, 2025) based evaluator: Accuracy for the closed-answer dataset (CSQA) and Rouge1 for the

open-answer dataset (VQA), comparing the LLM's answer a (generated from p') to the ground truth. ***Lower similarity scores indicate better privacy, while higher accuracy/Rouge1 scores indicate better utility***.

Comparative Baselines. We employ three competitive approaches as baselines: DP-Prompt (Utpala et al., 2023), a strong baseline leveraging zero-shot prompt learning on LLMs (GPT-3.5, Llama-3.1-8B (Meta, 2024), and FLAN-T5-Base (Chung et al., 2022)); DP-Paraphrase (Mattern et al., 2022), utilizing a GPT-2 model fine-tuned on SNLI; and DP-MLM (Meisenbacher et al., 2024b), based on a RoBERTa-Base masked language model.



(a) Rouge1 (GPT-3.5) (b) Rouge1 (Llama)

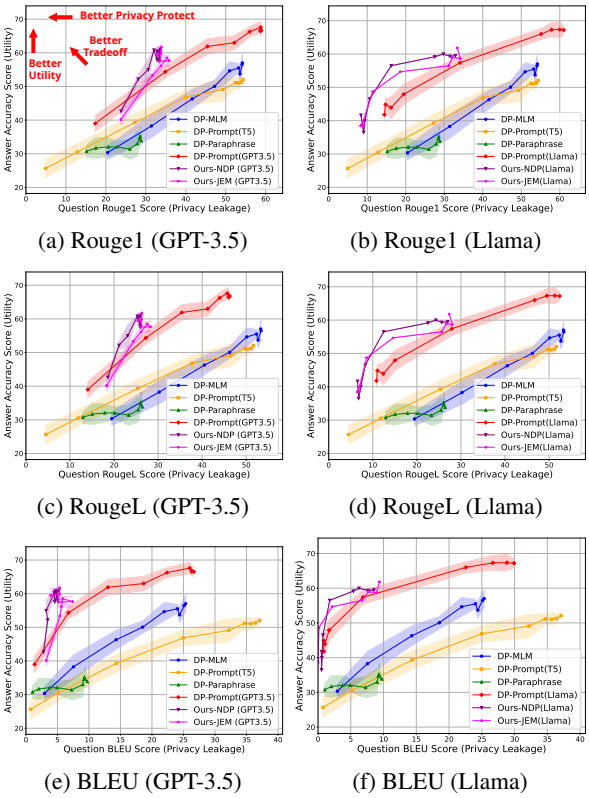
(c) RougeL (GPT-3.5) (d) RougeL (Llama)

(e) BLEU (GPT-3.5) (f) BLEU (Llama)

Figure 2: Privacy-utility trade-off for baselines and DP-GTR on open-answer PFL-DocVQA (VQA) dataset. The left column presents GPT-3.5 results, and the right column shows Llama-3.1-8B results. Refer to the x-axis label for specific measurement metrics.

DP-GTR Settings. We use GPT-3.5 (black-box) and Llama-3.1-8B (white-box) as underlying models for DP-GTR. Both the number of group text rewritings and private keywords are set to 10. For private keyword release in Stage-2, we implement a non-DP post-processing method (***DP-GTR-NDP***) and a differentially private JointEM mechanism (***DP-GTR-JEM***). In Stage-1, paraphrasing is controlled by temperature (black-box) and privacy budget ϵ (white-box), with nine values tested: $T \in \{0.1, 0.15, \ldots, 1.5\}$. Corresponding ϵ values for the white-box model are calculated based on temperature and pre-clipped sensitivity (see Appendix Section B). Stage-3 prompt generation uses a temperature of 0 (or equivalently low) without a DP mechanism.

Evaluation Repetitions. All experiments were repeated five times, and the reported results are the mean values with standard deviations (displayed as shaded areas in Figures 2 and 3).



(a) Rouge1 (GPT-3.5) (b) Rouge1 (Llama)

(c) RougeL (GPT-3.5) (d) RougeL (Llama)

(e) BLEU (GPT-3.5) (f) BLEU (Llama)

Figure 3: Privacy-utility trade-off for baselines and DP-GTR on close-answer Commonsense QA (CSQA) dataset. The left column presents GPT-3.5 results, and the right column shows Llama-3.1-8B results. Refer to the x-axis label for specific measurement metrics.

5.3 Results on Open-answer VQA Dataset

Figure 2 shows the results on the VQA dataset. DP-GTR (GPT-3.5) achieves a superior privacy-utility trade-off, consistently offering better privacy than DP-Prompt (GPT-3.5) at comparable utility, and sometimes higher utility at lower privacy. This validates our one-shot ICL utility design. With Llama, DP-GTR also maintains the best trade-off as temperature increases. DP-GTR-JEM, due to noisy keyword release, shows slightly higher privacy leakage compared to DP-GTR-NDP, but

both provide strong privacy. Other baselines (DP-Prompt(T5), DP-MLM, DP-Paraphrase) achieve high privacy but at the cost of unacceptably low utility, demonstrating the limitations of prior evaluations relying on simplistic semantic analysis for utility in current LLM-based QA systems.

5.4 Results on Close-answer CSQA Dataset

Figure 3 shows results on the CSQA dataset. DP-GTR converges faster and achieves a superior privacy-utility trade-off than baselines, generally outperforming them at equivalent privacy levels. While DP-Prompt (GPT-3.5 and Llama) shows higher utility in some cases, this comes at the cost of unacceptable privacy leakage. We identify a *Rapid Equilibrium Deterioration Interval (REDI)*, where privacy degrades sharply with minor utility gains. DP-Prompt's REDI is wide and discrete (16-45% for GPT-3.5, 20-55% for Llama on question Rouge1), making parameter tuning difficult. DP-GTR mitigates this, converging around a question Rouge1 of 30% and utility of 55-60%, achieving a more stable and robust trade-off. The early convergence of DP-GTR on CSQA, a dataset with strong logical coherence, indirectly confirms the effectiveness of our privacy keyword suppression.

5.5 Non-Uniform Rewriting Strategies

Beyond uniform temperature or epsilon settings, we investigated the impact of *non-uniform* rewriting strategies during group text rewriting. We conducted 10 rewriting tasks using DP-Prompt (Utpala et al., 2023) and our method (DP-GTR) on the GPT-3.5 model, with temperatures T ranging from 0.5 to 1.5 in increments of 0.1. Table 1 illustrates that our method achieves a favorable privacy-utility trade-off. For VQA, DP-GTR-NDP reduces privacy leakage from 56.91% to 43.94% while incurring a 2.69% utility loss. For CSQA, privacy leakage decreases from 56.91% to 43.94%, with a corresponding 8.55% utility loss. The total privacy budget in this non-uniform setting is $\sum_{i=1}^{10} \varepsilon_i$, where $\varepsilon_i = \frac{2n\Delta u}{T_i}$, $T_i \in \{0.5, 0.6, \ldots, 1.5\}$, and n is the number of tokens in the i-th generated text.

5.6 Generalizable Plug-in Framework

A significant contribution of this work is the development of a generalizable framework that functions as a plug-in, compatible with any existing paraphrasing method. The modular design, indicated by the blue arrow representing Stage-1 in Figure 1, enables the replacement of our DP-based text rewriting component with alternative paraphrasing techniques. To demonstrate this generalizability, we integrated the strong baseline method, DP-Prompt (Utpala et al., 2023), prior to our method, using the same base models (GPT-3.5 and Llama) in sequence. Thus, the results presented in Figures 2 and 3 also validate the plug-in capabilities and efficiency of our framework.

Table 1: Performance of rewriting under different temperatures on the VQA and CSQA datasets. Values are reported as mean (standard deviation) over five runs, for both Question (Privacy Leakage) and Answer (Utility).

Open-answer VQA Results				
Methods	**Question (Privacy Leakage)**			**Answer (Utility)**
	Rouge1	**RougeL**	**BLEU**	**Rouge1**
DP-Prompt	56.91 (18.0)	51.54 (16.8)	23.24 (13.3)	45.87 (0.0)
NDP	43.94 (1.3)	40.22 (1.1)	13.40 (0.6)	43.18 (1.3)
JEM	47.28 (0.8)	44.15 (0.5)	20.98 (0.5)	43.07 (0.7)
Close-answer CSQA Results				
Methods	**Question (Privacy Leakage)**			**Answer (Utility)**
	Rouge1	**RougeL**	**BLEU**	**Accuracy**
DP-Prompt	49.73 (13.5)	38.66 (10.6)	19.18 (9.4)	62.65 (8.3)
NDP	30.01 (0.3)	22.87 (0.2)	4.61 (0.1)	54.10 (1.0)
JEM	35.11 (1.1)	27.35 (1.2)	8.33 (0.8)	53.60 (1.7)

6 Ablation Study

The ablation experiments are conducted on two additional configurations: one with references but without keywords, and another with keywords but without references. The configuration is GPT-3.5 and DocVQA. The results are shown in Tab. 2.

Table 2: Question (Privacy Leakage) and Answer (Utility) ablation study. Subscripts indicate the difference between *without Reference/Keywords* and Ours-NDP. The privacy and utility metric is Rouge-1 (%) between final prompt and original prompt, and Rouge-1 (%) between query answer and ground-truth answer.

Temperature	$T = 0.1$	$T = 0.25$	$T = 0.5$	$T = 1.0$
Question (Privacy Leakage: Rouge-1)				
Ours-NDP	53.21	53.05	48.91	42.46
w/o Reference	$7.74_{\downarrow 45.47}$	$8.12_{\downarrow 44.93}$	$8.45_{\downarrow 40.46}$	$9.44_{\downarrow 33.02}$
w/o Keywords	$50.11_{\downarrow 3.10}$	$49.36_{\downarrow 3.69}$	$48.05_{\downarrow 0.86}$	$47.05_{\uparrow 4.59}$
Answer (Utility: Rouge-1)				
Ours-NDP	55.31	54.51	47.89	34.03
w/o Reference	$1.01_{\downarrow 54.30}$	$1.04_{\downarrow 53.47}$	$1.13_{\downarrow 46.76}$	$1.57_{\downarrow 32.46}$
w/o Keywords	$21.61_{\downarrow 33.70}$	$23.08_{\downarrow 31.43}$	$22.76_{\downarrow 25.13}$	$24.58_{\downarrow 9.45}$

Experimental results indicate that using either isolated privacy keywords or the reference alone yields significantly lower performance than our approach. Removing the reference substantially impairs the model's ability to generate valid questions, where utility's Rouge-1 score is no more than 1%. Omitting private keywords disrupts the trade-off, resulting in slight privacy leakage that is dispropor-

Table 3: The results of the adversarial attack. The Rouge-1 (%) scores measuring question similarity between the original and DP-GTR-processed prompts (**Ours**), as well as under Static and Adaptive attacks, across different temperatures for both NDP and JEM. Subscripts indicate change relative to ours.

Temperature	$T = 0.1$	$T = 0.25$	$T = 0.5$	$T = 1.0$
Ours-NDP	53.19	53.04	48.95	42.43
Static Attack	$52.27_{\downarrow 0.92}$	$52.37_{\downarrow 0.67}$	$48.31_{\downarrow 0.64}$	$40.21_{\downarrow 2.22}$
Adaptive Attack	$49.87_{\downarrow 3.32}$	$49.54_{\downarrow 3.50}$	$45.01_{\downarrow 3.94}$	$36.93_{\downarrow 5.50}$
Ours-JEM	53.50	53.77	50.92	46.17
Static Attack	$51.87_{\downarrow 1.63}$	$52.67_{\downarrow 1.10}$	$50.50_{\downarrow 0.42}$	$43.68_{\downarrow 2.49}$
Adaptive Attack	$49.60_{\downarrow 3.90}$	$50.48_{\downarrow 3.29}$	$46.90_{\downarrow 4.02}$	$39.77_{\downarrow 6.40}$

tionate to the utility loss. Optimal performance is achieved only when both elements are combined. We attribute this to the fact that, although privacy keywords serve as suppression targets, LLMs can leverage these keywords to produce effective paraphrases, thereby enhancing utility.

7 Adversarial Attack

Based on the threat model in Section 3, we design two adversarial experiments: **Static** and **Adaptive** attacks. In the static setting, the attacker is unaware of DP-GTR and directly attempts to recover the original question from cloud-received paraphrases. In the adaptive setting, the attacker fully understands DP-GTR, replicates its framework, and leverages privacy consensus keywords to infer the original question. In Table 3, using GPT-3.5 and DocVQA as examples, we report the Rouge-1 (%) score between the original and DP-GTR-processed prompts under different releasing strategies, along with the corresponding Δ for each attack. See detailed attack settings in Appendix D.

The results of adversarial attacks demonstrate that DP-GTR offers strong robustness against both static and adaptive threats, with no observed increase in question Rouge-1 similarity, indicating no privacy leakage. These findings underscore effectiveness in safeguarding against adversarial risks.

8 Conclusion

This paper proposes DP-GTR, a novel three-stage local differential privacy (LDP) framework that leverages differentially private paraphrasing and the composition theorem through group text rewriting to enhance the privacy-utility trade-off. DP-GTR is the first approach, to our knowledge, to apply in-context learning for LDP prompt privatization and to connect global and local DP mechanisms via grouped paraphrased text. Furthermore, our framework is generalizable and compatible with any existing paraphrasing technique. Evaluations on open- and closed-answer QA datasets (DocVQA and Commonsense QA), simulating real-world LLM application scenarios, demonstrate that DP-GTR achieves a significantly superior privacy-utility trade-off compared to existing state-of-the-art methods. With the rapidly increasing adoption of LLMs, DP-GTR provides a practical, robust and readily deployable solution for mitigating the risk of user prompt privacy leakage.

9 Limitations

The primary limitation of our work is that LLMs may sometimes fail to follow instructions, potentially leading to privacy leakage or an inability to learn from one-shot utility exemplars. In future work, an important direction is to shift control from outlier prompting to an internal LLM generation configuration. This approach will fundamentally address the issue of prompt failure in LLMs.

Another limitation arises from computing resource constraints, which led us to choose the Llama-3.1-8B open-source model. This model may not effectively learn from its prompt, resulting in relatively poor performance. We believe that with a more capable open-source model, the framework would perform better.

10 Acknowledgment

This work was supported in part by the CAHSI–Google Institutional Research Program (IRP) Grant and by the National Science Foundation (NSF) under grants CCF-2447834, HSI-2225229, CNS-2231519, and DUE-2225229.

References

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318.

Mário Alvim, Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Anna Pazii. 2018. Local differential privacy on metric spaces: optimizing the trade-off with utility. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pages 262–267. IEEE.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Ricardo Silva Carvalho, Theodore Vasiloudis, Oluwaseyi Feyisetan, and Ke Wang. 2023. Tem: High utility metric differential privacy on text. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*, pages 883–890. SIAM.

Sai Chen, Fengran Mo, Yanhao Wang, Cen Chen, Jian-Yun Nie, Chengyu Wang, and Jamie Cui. 2023a. A customized text sanitization mechanism with differential privacy. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5747–5758, Toronto, Canada. Association for Computational Linguistics.

Yu Chen, Tingxin Li, Huiming Liu, and Yang Yu. 2023b. Hide and seek (has): A lightweight framework for prompt privacy protection. *Preprint*, arXiv:2309.03057.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *arXiv preprint*.

John C Duchi, Michael I Jordan, and Martin J Wainwright. 2013. Local privacy and statistical minimax rates. In *2013 IEEE 54th annual symposium on foundations of computer science*, pages 429–438. IEEE.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pages 265–284. Springer.

Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407.

Kennedy Edemacu and Xintao Wu. 2025. Privacy preserving prompt engineering: A survey. *ACM Computing Surveys*, 57(10):1–36.

Oluwaseyi Feyisetan, Borja Balle, Thomas Drake, and Tom Diethe. 2020. Privacy-and utility-preserving textual analysis via calibrated multivariate perturbations. In *Proceedings of the 13th international conference on web search and data mining*, pages 178–186.

Fengyu Gao, Ruida Zhou, Tianhao Wang, Cong Shen, and Jing Yang. 2024. Data-adaptive differentially private prompt synthesis for in-context learning. *arXiv preprint arXiv:2410.12085*.

Jennifer Gillenwater, Matthew Joseph, Andres Munoz, and Monica Ribero Diaz. 2022. A joint exponential mechanism for differentially private top-k. In *International Conference on Machine Learning*, pages 7570–7582. PMLR.

Junyuan Hong, Jiachen T Wang, Chenhui Zhang, LI Zhangheng, Bo Li, and Zhangyang Wang. 2024. Dp-opt: Make large language model your privacy-preserving prompt engineer. In *The Twelfth International Conference on Learning Representations*.

Timour Igamberdiev and Ivan Habernal. 2023. Dp-bart for privatized text rewriting under local differential privacy. *arXiv preprint arXiv:2302.07636*.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1. Minneapolis, Minnesota.

Mike Lewis. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Zeren Luo, Zifan Peng, Yule Liu, Zhen Sun, Mingchen Li, Jingyi Zheng, and Xinlei He. 2025. Unsafe llm-based search: Quantitative analysis and mitigation of safety risks in ai web search. *arXiv preprint arXiv:2502.04951*.

Justus Mattern, Benjamin Weggenmann, and Florian Kerschbaum. 2022. The limits of word level differential privacy. *arXiv preprint arXiv:2205.02130*.

Frank McSherry and Kunal Talwar. 2007. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 94–103. IEEE.

Casey Meehan, Khalil Mrini, and Kamalika Chaudhuri. 2022. Sentence-level privacy for document embeddings. *arXiv preprint arXiv:2205.04605*.

Stephen Meisenbacher, Maulik Chevli, and Florian Matthes. 2024a. 1-diffractor: Efficient and utility-preserving text obfuscation leveraging word-level metric differential privacy. *arXiv preprint arXiv:2405.01678*.

Stephen Meisenbacher, Maulik Chevli, Juraj Vladika, and Florian Matthes. 2024b. Dp-mlm: Differentially private text rewriting using masked language models. *arXiv preprint arXiv:2407.00637*.

Stephen Meisenbacher and Florian Matthes. 2024. Just rewrite it again: A post-processing method for enhanced semantic similarity and privacy preservation of differentially private rewritten text. In *Proceedings of the 19th International Conference on Availability, Reliability and Security*, pages 1–11.

Meta. 2024. Introducing llama 3.1: Our most capable models to date.

Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2007a. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 75–84.

Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2007b. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 75–84.

OpenAI. 2025. Gpt-3.5 turbo.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Zifan Peng, Yule Liu, Zhen Sun, Mingchen Li, Zeren Luo, Jingyi Zheng, Wenhan Dong, Xinlei He, Xuechao Wang, Yingjie Xue, et al. 2025. Jalmbench: Benchmarking jailbreak vulnerabilities in audio language models. *arXiv preprint arXiv:2505.17568*.

Natalia Ponomareva, Jasmijn Bastings, and Sergei Vassilvitskii. 2022. Training text-to-text transformers with privacy guarantees. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2182–2193.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *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 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Xinyu Tang, Richard Shin, Huseyin A Inan, Andre Manoel, Fatemehsadat Mireshghallah, Zinan Lin, Sivakanth Gopi, Janardhan Kulkarni, and Robert Sim. 2023. Privacy-preserving in-context learning with differentially private few-shot generation. *arXiv preprint arXiv:2309.11765*.

Rubèn Tito, Khanh Nguyen, Marlon Tobaben, Raouf Kerkouche, Mohamed Ali Souibgui, Kangsoo Jung, Joonas Jälkö, Vincent Poulain D'Andecy, Aurelie Joseph, Lei Kang, et al. 2024. Privacy-aware document visual question answering. In *International Conference on Document Analysis and Recognition*, pages 199–218. Springer.

Saiteja Utpala, Sara Hooker, and Pin Yu Chen. 2023. Locally differentially private document generation using zero shot prompting. *arXiv preprint arXiv:2310.16111*.

Tianhao Wang, Jeremiah Blocki, Ninghui Li, and Somesh Jha. 2017. Locally differentially private protocols for frequency estimation. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 729–745.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Tong Wu, Ashwinee Panda, Jiachen T Wang, and Prateek Mittal. 2023. Privacy-preserving in-context learning for large language models. *arXiv preprint arXiv:2305.01639*.

Zekun Xu, Abhinav Aggarwal, Oluwaseyi Feyisetan, and Nathanael Teissier. 2020. A differentially private text perturbation method using a regularized mahalanobis metric. *arXiv preprint arXiv:2010.11947*.

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. 2024. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, 4(2):100211.

Xiang Yue, Minxin Du, Tianhao Wang, Yaliang Li, Huan Sun, and Sherman SM Chow. 2021. Differential privacy for text analytics via natural text sanitization. *arXiv preprint arXiv:2106.01221*.

Chunyan Zheng, Keke Sun, Wenhao Zhao, Haibo Zhou, Lixin Jiang, Shaoyang Song, and Chunlai Zhou. 2024. Locally differentially private in-context learning. *arXiv preprint arXiv:2405.04032*.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2024. Rose doesn't do that: Boosting the safety of instruction-tuned large language models with reverse prompt contrastive decoding. *arXiv preprint arXiv:2402.11889*.

A DP-guaranteed Paraphrasing Proof

Proof. Let D and D' represent two arbitrary documents, and $\mathbf{u}, \mathbf{u}' \in \mathbb{R}^{|V|}$ denote the associated logit vectors. Set all the logits lie within the interval $[b_{min}, b_{max}]$ under sensitivity clipping. For a given token $v \in V$, let i denote its position index and u_i represent its corresponding logit from $\mathbf{u}$. Consequently, we obtain (Utpala et al., 2023),

$$
\begin{aligned}
\frac{\Pr[M(D)=v]}{\Pr[M(D')=v]} &= \frac{\frac{\exp(\frac{u_i}{T})}{\sum_{j=1}^{|V|}\exp(\frac{u_j}{T})}}{\frac{\exp(\frac{u'_i}{T})}{\sum_{j=1}^{|V|}\exp(\frac{u'_j}{T})}} \\
&= \frac{\exp(\frac{u_i}{T})\sum_{j=1}^{|V|}\exp(\frac{u'_j}{T})}{\exp(\frac{u'_i}{T})\sum_{j=1}^{|V|}\exp(\frac{u_j}{T})} \\
&= \exp\left(\frac{u_i-u'_i}{T}\right)\frac{\sum_{j=1}^{|V|}\exp(\frac{u'_j}{T})}{\sum_{j=1}^{|V|}\exp(\frac{u_j}{T})} \\
&\le \exp\left(\frac{b_{max}-b_{min}}{T}\right)\exp\left(\frac{b_{max}-b_{min}}{T}\right) \\
&\le \exp\left(\frac{2(b_{max}-b_{min})}{T}\right).
\end{aligned}
$$

Algorithm 2 DP-Prompt Algorithm (Utpala et al., 2023)

Require: Language model LM, Private document D, Private Budget ϵ, Prompt template T, Logit bounds $\mathbf{b} \in \mathbb{R}^{|\mathcal{V}|}$ with $b_{min} \le u_v \le b_{max}$, Number of generated tokens n

Ensure: Sanitized text P

1: **Generate Prompt:** Construct an initial context $\tilde{C}$ from $\{D, T\}$ and tokenize it
2: LM $\leftarrow$ `clipLogits`$(u, \mathbf{b})$
3: Temp $\leftarrow \left(\frac{2\left(b_{max}-b_{min}\right)}{\epsilon}\right)$
4: LM $\leftarrow$ `setTemperature`(Temp)
5: **for** $i = 1$ to n **do**
6: $\quad u \leftarrow \text{LM}(\tilde{C})$
7: $\quad v \leftarrow$ `ExponentialMechanism`(u)
8: $\quad P \leftarrow P \cup \{v\}, \quad \tilde{C} \leftarrow \tilde{C} \cup \{v\}$
9: **end for**
10: **Output:** Detokenize(P)

B Epsilon and Sensitivity Setting

The sensitivity bound is the other critical theoretical parameter. Following prior work (Igamberdiev and Habernal, 2023), we adopt a pre-clipping strategy. Specifically, we randomly sample 1,000 examples from the CSQA training dataset and perform the DP-Prompt paraphrasing task while recording all logits. We then compute the mean (μ) and standard deviation (σ), and define the sensitivity bound as $(\mu, \mu + 4\sigma)$ to better preserve high-value logits (Meisenbacher et al., 2024b). We also detail the exact logits clipping ($[b_{min}, b_{max}]$) range used in Appendix B.

Table 4: Logits clipping bounds (b_{min} and b_{max}) for different DP-based models.

Model	b_{min}	b_{max}
DP-MLM (RoBERTa)	-3.2093	16.3048
DP-Paraphrase (GPT-2)	-96.8525	-8.7477
DP-Prompt (Flan-T5)	-19.2271	7.4832
Ours (Llama-3.1-8B)	22.9443	32.6274

The corresponding ϵ is computed using the alignment target temperature with the formula $(\frac{2(b_{\max}-b_{\min})}{T})$. See the Table 5 for detailed values.

Table 5: Epsilon values for different methods introduced in Section 5.2 across temperatures (**T**).

T	**DP-MLM**	**DP-Paraphrase**	**DP-Prompt**	**Ours**
	RoBERTa	**GPT-2**	**FLAN-T5**	**Llama**
0.10	390.0	1760.0	534.2	194.0
0.15	260.0	1173.3	356.1	129.3
0.20	195.0	880.0	267.1	97.0
0.25	156.0	704.0	213.7	77.6
0.50	78.0	352.0	106.8	38.8
0.75	52.0	234.7	71.2	25.9
1.00	39.0	176.0	53.4	19.4
1.25	31.2	140.8	42.7	15.5
1.50	26.0	117.3	35.6	12.9

Additionally, in the setting of Ours-JEM, ε_2 for JointEM is set to **2**.

C Computational Costs

DP-GTR is an inference-side, single-prompt privacy protection approach that demonstrates lower overall computational resource consumption compared to methods requiring model training (Hong et al., 2024). The primary source of computational overhead stems from the grouping step involved in paraphrasing. Below, we provide a detailed computational analysis.

- **Computation Efficiency:** DP-GTR is designed to support parallel generation of paraphrased texts, which can substantially reduce inference time from $O(n+1)$ to $O(2)$ when there are sufficient computational resources.
- **Memory Efficiency:** When parallel processing is not feasible, GTR can operate in serial mode. In this case, only the sentence with the lowest perplexity and the corresponding privacy keyword votes are retained in memory, thereby minimizing memory consumption from $O(n)$ to $O(1)$.

- **User-End Efficiency:** For user-end environments with low throughput and limited computational resources, DP-GTR does not require model training and offers particular advantages in both memory and computation efficiency.

While we acknowledge that DP-GTR introduces additional computational overhead relative to baseline inference, we argue that this trade-off is justified. The enhanced privacy protection offered by DP-GTR is particularly important for practical deployments, given the increasing privacy risks associated with large language models.

D Adversarial Attack

We consider a threat model in which the attacker has full capability at the cloud end at Preliminaries 3. Accordingly, we detail both static and adaptive attack strategies. In the static attack, the adversary directly prompts the LLM to reconstruct the original question using the template shown in Template 1. For the adaptive attack, the adversary replicates the entire workflow, including candidate paraphrase generation, private consensus keyword extraction, and reference selection, and then combines this information to recover the original question. The adaptive prompt recovery template is provided in Template 2.

Static Prompt Template

Given a paraphrased question: <$\hat{P}$>
Convert to the original question:

Adaptive Prompt Template

Refer the paraphrased question to generate the original question: <$\hat{P}_{low}$>
Use following tokens: <$\hat{w}_1, \hat{w}_2, ..., \hat{w}_k$>
Generated original question:

E Explainable Case

To illustrate our approach, we present an explainable case below, which is forked from our open-source repository and generated by DP-GTR-NDP (GPT-3.5). The **Document** is the original question, followed by **10 DP-guaranteed paraphrases**. From these, we extract the **Private Keywords** and identify the lowest-perplexity **Reference**, ultimately producing a differentially private, protected document.

DP-GTR Example

Document:
`In which year, john f. kennedy was assassinated?`

10 DP-guaranteed Paraphrases:
- In what year was John F. Kennedy killed?
- What year was John F. Kennedy assassinated?
- During which year did John F. Kennedy get assassinated?
- In what year was John F. Kennedy killed?
- What year did John F. Kennedy get assassinated?
- In what year was John F. Kennedy killed?
- What year did John F. Kennedy get assassinated?
- In what year was John F. Kennedy killed?
- What year did John F. Kennedy get shot?
- In what year did John F. Kennedy get killed?

Private Keywords:
`What, get, Kennedy, killed, In, John, assassinated, F., year, During`

Reference:
`What year was John F. Kennedy assassinated?`

Final Generated Document:
`When did the tragic event occur involving the 35th President of the United States?`