DYNTEXT: Semantic-Aware Dynamic Text Sanitization for Privacy-Preserving LLM Inference

Juhua Zhang[1], Zhiliang Tian[1*], Minghang Zhu[3], Yiping Song[2], Taishu Sheng[1], Siyi Yang[1], Qiunan Du[1], Xinwang Liu [1], Minlie Huang[4], Dongsheng Li[1*]

[1]College of Computer Science and Technology, National University of Defense Technology, Hunan, China,
[2]College of Science, National University of Defense Technology, Hunan, China,
[3]College of Computer Science and Technology, Shandong University, Qingdao, China,
[4]The CoAI Group, DCST, BNRist, Tsinghua University, Beijing, China

{zhangjuhua23, tianzhiliang, songyiping, sts, yangsiyi22, duqiunan, xinwangliu, dsli}@nudt.edu.cn,
zhuminghang0423@gmail.com, aihuang@tsinghua.edu.cn

Abstract

LLMs face privacy risks when handling sensitive data. To ensure privacy, researchers use differential privacy (DP) to provide protection by adding noise during LLM training. However, users may be hesitant to share complete data with LLMs. Researchers follow local DP to sanitize the text on the *user side* and feed non-sensitive text to LLMs. The sanitization usually uses a fixed non-sensitive token list or a fixed noise distribution, which induces the risk of being attacked or semantic distortion. We argue that the token's protection level should be adaptively adjusted according to its semantic-based information to balance the privacy-utility trade-off. In this paper, we propose DYNTEXT, an LDP-based Dynamic Text sanitization for privacy-preserving LLM inference, which dynamically constructs semantic-aware adjacency lists of sensitive tokens to sample non-sensitive tokens for perturbation. Specifically, DYNTEXT first develops a semantic-based density modeling under DP to extract each token's density information. We propose token-level smoothing sensitivity by combining the idea of global sensitivity (GS) and local sensitivity (LS), which dynamically adjusts the noise scale to avoid excessive noise in GS and privacy leakage in LS. Then, we dynamically construct an adjacency list for each sensitive token based on its semantic density information. Finally, we apply the replacement mechanism to sample non-sensitive, semantically similar tokens from the adjacency list to replace sensitive tokens. Experiments show that DYNTEXT excels strong baselines on three datasets.

1 Introduction

LLMs demonstrated exceptional capabilities in NLP tasks, particularly with closed-source LLMs like GPT-4 (Open, 2023) that exclusively provide online inference services. However, directly submitting text containing sensitive information to those LLMs poses significant privacy risks (Huang et al., 2023). To ensure privacy protection, A provable theoretical guarantee is crucial. DP (Dwork et al., 2014) formally defines and quantifies privacy. Consequently, most researchers apply DP to LLMs to safeguard privacy (Edemacu and Wu, 2024).

To achieve DP, methods like DP-SGD (Abadi et al., 2016) and PATE (Papernot et al., 2016), mainly focus on adding calibrated noise to the model or input representations during the training so that sensitive user data are hardly inferred from the trained model. Users need to send their data to LLMs for training under the DP framework with noise. However, they may hesitate to share their complete data due to privacy concerns, fearing that LLMs may not be fully trustworthy or that an intermediary eavesdropper could compromise sensitive information (Lyu et al., 2020). To address the above issues, LDP (Duchi et al., 2013) introduces a new scenario with two phases: local processing and LLM training/inference. Local processing occurs on the user side, which can access and process the private data to protect them. The protected data are then transmitted to LLMs for training or inference. Typically, these local processing methods generate perturbed text by replacing the tokens (e.g., words or n-grams) in the private text with new non-sensitive tokens (Feyisetan et al., 2019; Qu et al., 2021). Specifically, some methods (Feyisetan et al., 2020; Li et al., 2025) inject calibrated noise with a DP guarantee into the original token embedding (high-dimensional vector) to generate a noisy embedding, then replace the original token with the token closest to the noisy embedding. However, token (i.e. text) embedding space is usually uneven and irregular since the text signals are too sparse and discrete to represent with dense embeddings so well (Yaghoobzadeh and Schütze, 2016; Yin and Shen, 2018; Zheng et al., 2023). DP-required noises are totally randomized within a regular distribution (i.e. Gaussian or Laplace). Applying a

*Corresponding Authors

Findings of the Association for Computational Linguistics: ACL 2025, pages 20243–20255
July 27 - August 1, 2025

DP required noises to the original token embedding sometimes leads to unexpected bias to damage the semantics.

To avoid the above problem, researchers propose replacing the original tokens by sampling new tokens from a pre-computed distribution. These methods, like SANTEXT+ (Yue et al., 2021) and CUSTEXT+ (Chen et al., 2023), leverage DP learning methods to sequentially replace sensitive words in text with new words, which are sampled from a fixed word list carrying non-sensitive words similar to the sensitive words. These methods are more reliable and interpretable, effectively avoiding unexpected bias caused by noise and thereby enhancing the practicality of the text. However, the fixed token list introduces predictable replacement patterns, making it easier for attackers to exploit this regularity of information to infer the original sensitive information (Tong et al., 2025).

To mitigate the above vulnerability for potential attacks (Song and Raghunathan, 2020), researchers introduce randomness in the non-sensitive tokens list to replace each sensitive token, avoiding potential attacks and strengthening defense against privacy threats (Tong et al., 2025; Fan et al., 2024). However, adding random perturbation to non-sensitive lists still has limitations. These methods often apply perturbations with the same distribution to all tokens, ignoring the sensitivity of each token. For tokens with low semantic sensitivity, overly strict privacy protection mechanisms may lead to unnecessary semantic loss, thus affecting the quality of the generated perturbed text.

To balance the privacy-utility trade-off, we argue that sanitization should consider the token's semantic-based information while maintaining anti-attack capabilities. So, we should integrate the token's semantic-based information with its non-sensitive token list under privacy protection (i.e. DP), enhancing the quality of the perturbed text and adaptively adjusting the list to resist attacks.

In this paper, we propose an LDP-based Dynamic Text (DYNTEXT)[1] sanitization mechanism for privacy-preserving LLM inference, which dynamically builds a semantic-aware adjacency list of sensitive tokens to sample non-sensitive tokens for perturbation. The adjacency list satisfies DP and is customized to each token's semantic density, with smaller lists in high-density areas and larger ones in low-density areas, which encourages the sampling of high-density tokens and assigning high noise to low-density tokens. Specifically, we first develop a semantic-based density information modeling module under DP to extract the density information of each token in the embedding space. This module employs the Gaussian noise to achieve DP and a token-level smoothing sensitivity mechanism by combining the idea of GS and LS to avoid excessive noise in GS and privacy leakage in LS. We then dynamically construct an adjacency list for each sensitive token based on noisy semantic-based density information, which adjusts the size of each sensitive token's non-sensitive adjacency token list. This strategy effectively preserves semantic information while resisting attacks. Finally, we employ a sensitive token replacement to sample non-sensitive similar tokens from the adjacency list and replace the sensitive token for perturbation.

Our contributions are as follows: (1) We propose DYNTEXT, an LDP-based dynamic text sanitization mechanism that replaces sensitive tokens based on semantic density, adaptively adjusting the protection level for a better privacy-utility trade-off. (2) We design a DP-compliant semantic-aware dynamic adjacency list adjusted by token density information, promoting sampling from high-density areas for semantic preservation and assigning high noise to low-density areas for privacy protection. (3) Experiments show that DYNTEXT excels in all baselines and achieves SOTA on three datasets.

2 Related Work

2.1 Privacy Protection in LLMs

The privacy protection lifecycle of LLMs includes training and inference phases. (1) Most of the previous work focuses on privacy protection during training, where DP reduces privacy risks by adding noise (Tholoniat et al., 2024; Wicker et al., 2024). ANADP (Li et al., 2024) allocates noise and privacy budgets based on the importance of the parameters. (2) Current research is gradually focusing on protecting input privacy during inference, addressing challenges through data anonymization (Yang et al., 2024) and text-to-text privatization (Li et al., 2025).

2.2 DP learning algorithm

DP implementations mainly use gradient or output perturbation techniques. (1) Gradient perturbation approaches modify training gradients. The DP-SGD framework (Abadi et al., 2016) applies gradi-

[1]The implementation is available at: `https://github.com/mhyt-ning/DYNTEXT`.

ent clipping followed by Gaussian noise injection to limit the influence of individual data points. Subsequent studies (Yue et al., 2023; Kurakin et al., 2023) refine these noise injection and clipping mechanisms to speed up convergence. Adaptive noise scheduling (Yang and Ma, 2024; Jiao et al., 2024) optimizes the approach by adjusting noise levels based on gradient sensitivity and selectively updating parameters to reduce noise accumulation. Output perturbation methods include PATE-based approaches (Yuan et al., 2024; Song et al., 2024; Tian et al., 2022; Papernot et al., 2016), which produce private labels by noisy voting from multiple teacher models, and objective perturbation (Pustozerova et al., 2023), which adds noise directly to the loss function to prevent gradient exposure.

2.3 Local Privacy Protection for LLMs

Recent advancements in local privacy preservation for LLMs reveal trade-offs between security and practicality. LDP approaches (MLDP (Feyisetan et al., 2020), SANTEXT+ (Yue et al., 2021)) introduce word/vector-level sanitization mechanisms that risk semantic distortion, while CUSTEXT+ (Chen et al., 2023) improves output quality at potential privacy costs. SnD (Mai et al., 2024)'s denoising pipelines reduce semantic distortion but introduce system complexity due to the need for additional model training. RANTEXT (Tong et al., 2025) applies LDP with dynamic random adjacency lists and knowledge distillation to enhance privacy (Lee et al., 2022). The above methods struggle to balance the privacy-utility trade-off. In contrast, our approach dynamically adjusts privacy protection based on semantic density information, achieving an effective balance.

3 Preliminaries

Differential Privacy (DP) (Dwork, 2006; Dwork et al., 2014)) is widely regarded as the gold standard for data privacy. Its definition is as follows:

Definition 3.1 (ε, δ)**-Differential Privacy.** Let $\varepsilon \geq 0$ and $\delta \in [0, 1]$. A randomized algorithm $\mathcal{M}$ is (ε, δ)-differentially private if for any two neighboring datasets D and D', which differ in only a single record, and for any set S of possible outputs:

$$Pr[\mathcal{M}(D) \in S] \leq e^{\varepsilon} Pr[\mathcal{M}(D') \in S] + \delta. \tag{1}$$

where ε upper bounds the privacy loss, and δ is the probability that this guarantee does not hold.

Local Differential Privacy (LDP) is a special case of DP in which the server is untrusted, and data privatization is performed on the client side.

Definition 3.2 (ε, δ)**-Local Differential Privacy.** Let $\varepsilon \geq 0$ and $\delta \in [0, 1]$. A randomized mechanism $\mathcal{M}$ is said to satisfy (ε, δ)-LDP if for any two possible inputs $x, x' \in D$ and any possible output $y \in Y$, the following condition holds:

$$Pr[\mathcal{M}(x) = y] \leq e^{\varepsilon} Pr[\mathcal{M}(x') = y] + \delta. \tag{2}$$

4 Methods

4.1 Overview

Our proposed **DYNTEXT** consists of three modules, as shown in Fig. 1: (1) **Semantic-based Density Information Modeling under DP** (§4.2) obtains the semantic-based density information of each token in the embedding space while satisfying DP; (2) **Dynamic Construction of Adjacency List** (§4.3) constructs an adjacency list with dynamically adjustable size based on the semantic-based density information. The list contains a set of non-sensitive tokens with semantics similar to the target-sensitive token, serving as candidates for replacing the target token; (3) **Private Token Replacement via Similarity** (§4.4) samples a new token from the adaptive adjacency list considering the similarity between the sensitive token and candidate tokens, and then replace the sensitive token to generate the non-sensitive text. In summary, we first obtain semantic-based density (§4.2) to construct the adjacency list for sensitive tokens (§4.3), and then sample a non-sensitive token from that list to replace the sensitive token (§4.4) to generate sanitized texts. The sanitized texts act as the input for downstream text generation tasks.

4.2 Semantic-based Density Information Modeling under DP

We model the semantic-based density information of each token in the semantic embedding space, which applies Gaussian noise to achieve DP (§4.2.1) and token-level smooth sensitivity mechanism to mitigate impacts of abnormal data (§4.2.2).

The density information is used to adjust the privacy protection degree for different tokens (details in §4.3), aiming to enhance protection in low-density areas while moderately relaxing it in high-density areas, thereby improving the practicality of the DP algorithm. This is because, as inspired by TEM (Carvalho et al., 2023), low-density areas in

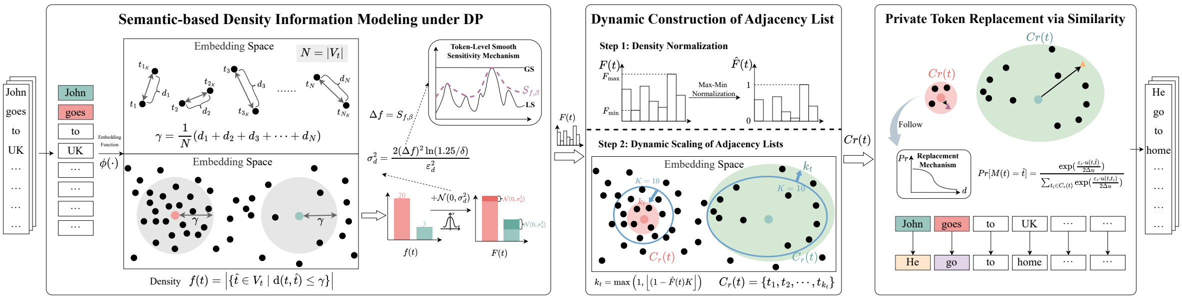


Figure 1: Overview of **DYNTEXT**. Given a sensitive text, DYNTEXT sanitizes it through three modules, executed sequentially: (1) **Semantic-based Density Information Modeling under DP** extracts each token's semantic-based density information in the embedding space and applies noise; (2) **Dynamic Construction of Adjacency List** builds an adjacency list for each token based on this density information; (3) Finally, **Private Token Replacement via Similarity** samples non-sensitive tokens from the adjacency list to replace sensitive tokens.

the embedding space typically correspond to rare tokens with fewer semantically similar words. Rare tokens are often sensitive because they have low entropy, high information content, and are more likely to represent entities. So, tokens in low-density areas are more vulnerable to privacy leakage and thus more sensitive. In contrast, high-density areas present lower privacy risks and sensitivity.

4.2.1 Density Calculation with Gaussian Mechanism

We obtain the target tokens' density information and apply the Gaussian mechanism for protection.

Density Calculation. We compute density information with three steps: (1) **Semantic distance.** In the N-dimensional embedding space $\mathbb{R}^N$, we first calculate the Euclidean distance between each token t and all tokens (including itself) in the token vocabulary V_t. Next, we identify the K-th closest token t_K to t and obtain their distance as follows:

$$d(t, t_K) = ||\phi(t) - \phi(t_K)||_2, \tag{3}$$

where the function $\phi : V_t \to \mathbb{R}^N$, maps each token to a vector in embedding space. The parameter K represents the default size of the adjacent list for a token $t \in V_t$. (2) **Density range.** We calculate a threshold γ as the density range of tokens. For each token $t \in V_t$, we compute the semantic distance $d(t, t_K)$; then, γ is defined as the average distance of tokens in V_t: $\gamma = \frac{1}{|V_t|}\sum_{t \in V_t} d(t, t_K)$. (3) **Density information.** We define the density information $f(t)$ of token t as the number of tokens $\hat{t}$ in V_t whose semantic distance to token t is less than or equal to the threshold γ:

$$f(t) = \left|\{\hat{t} \in V_t \mid \mathrm{d}(t, \hat{t}) \leq \gamma\}\right|. \tag{4}$$

The $f(t)$ reflects the number of neighboring tokens within a certain range around the target token t, making it a valuable measure of its density information in the embedding space. The threshold γ controls the range of the local neighborhood, ensuring that only tokens semantically close enough to the target token are considered in the density calculation, thereby defining the "local dense area".

Gaussian Mechanism. To prevent density information from leaking information (i.e. semantic density) of sensitive tokens, we add calibrated Gaussian noise (Bu et al., 2020) to the density information $f(t)$ of the token $t \in V_t$ to satisfy DP, as $F(t) = f(t) + \mathcal{N}(0, \sigma_d^2)$. It satisfies (ε_d, δ)-DP for $\varepsilon_d \geq 0$, where $\mathcal{N}(0, \sigma_d^2)$ represents Gaussian noise with mean 0 and variance σ_d^2. The variance of Gaussian noise σ_d^2 is determined by the privacy budget parameter ε_d and the sensitivity Δf:

$$\sigma_d^2 = \frac{2(\Delta f)^2 \ln(1.25/\delta)}{\varepsilon_d^2}. \tag{5}$$

4.2.2 Token-Level Smooth Sensitivity Mechanism

To reduce noise amplitude and mitigate privacy leaks from sensitivity fluctuations, we propose the token-level smooth sensitivity for more stable and controlled noise addition at the token level.

Existing methods mainly determine the noise scale via global and local sensitivity. GS (Iooss and Lemaître, 2015) represents the maximum change of the query function, which takes input data and returns statistical information, across all possible inputs. LS (Nguyen et al., 2024) measures the change based on the specific data. During density calculation, for any token t, density information $f(t)$ of token t acts as the query function f here (as shown in Eq. 4). The local sensitivity $LS_f(t)$ of the query function f is defined as: $LS_f(t) = \max_{\hat{t}\in Cr(t)} |f(t) - f(\hat{t})|$, where $\hat{t} \in C_r(t)$ is a token in the adjacency list of t. The global sensitivity GS_f is defined as: $GS_f = \max_{t\in V_t}(LS_f(t))$. However, both of the above sensitivities have their limitations. GS is based on the worst-case estimate across all possible input tokens, often resulting in excessive noise due to its conservatism. In contrast, LS dynamically adjusts the noise amplitude based on the information of each input token. However, this also means that the noise amplitude itself could potentially leak the privacy of the input token, and LS alone cannot satisfy the requirements of DP[2].

Hence, we propose a token-level smooth sensitivity mechanism that combines global and local sensitivity ideas at the token level. We use a "smoothed" approximation of LS to adjust the noise scale and prevent leaks of sensitive information. Specifically, we use the β-smooth sensitivity $S_{f,\beta}(t)$ (defined in Eq. 6) when adding noise to the token t's density information. For a token $t \in V_t$, t's adjacent token $\hat{t} \in C_r(t)$, $S_{f,\beta}(t)$ has two parts:

- $LS_f(\hat{t})$ represents the LS of $\hat{t}$.
- $e^{-\beta d(t,\hat{t})}$ is an exponential decay function, where $d(t,\hat{t})$ is the Euclidean distance (Eq. 3) between adjacent tokens. β is defined as $\frac{\varepsilon_d}{2\log(2/\delta)}$, controlling the impact of distance.

With Eq. 6, the LS is smoothed: (1) for each token $\hat{t} \in Cr(t)$, its local sensitivity $LS_f(\hat{t})$ is scaled using the exponential decay function $e^{-\beta d(t,\hat{t})}$; (2) the scaled maximum value $\max_{\hat{t}\in Cr(t)}(LS_f(\hat{t}) \cdot e^{-\beta d(t,\hat{t})})$ is selected as the smooth sensitivity $S_{f,\beta}(t)$ of the target token t:

$$S_{f,\beta}(t) = \max_{\hat{t}\in C_r(t)} (LS_f(\hat{t}) \cdot e^{-\beta d(t,\hat{t})}). \tag{6}$$

As the distance $d(t,\hat{t})$ between adjacent tokens increases, the decay function $e^{-\beta d(t,\hat{t})}$ decreases rapidly, thereby lowering the value of $LS_f(\hat{t}) \cdot e^{-\beta d(t,\hat{t})}$. Since $S_{f,\beta}(t)$ is the maximum value of $LS_f(\hat{t}) \cdot e^{-\beta d(t,\hat{t})}$, tokens closer to the target token are more likely to contribute to the maximum value than distant tokens. So, $S_{f,\beta}(t)$ is more sensitive to changes in closer tokens, allowing it to better preserve semantic features while avoiding excessive interference. Additionally, a larger β accelerates the decay of $e^{-\beta d(t,\hat{t})}$, emphasizing neighboring tokens while reducing the influence of distant ones; conversely, a smaller β slows the decay of $e^{-\beta d(t,\hat{t})}$, allowing distant tokens to contribute more, thereby enhancing privacy protection.

The benefit of the above method is twofold: (1) Compared to GS, our proposed smooth sensitivity incorporates LS to dynamically adjust the noise amplitude for each input token, reducing the noise amplitude and thus improving the model performance. (2) Compared to LS, our proposed smooth sensitivity mitigates the fluctuations in the sensitivity of individual data, weakening the impact of outliers, thereby ensuring that the sensitivity satisfies DP. Since the smooth sensitivity calculation of the target token t incorporates LS of all adjacent tokens, the adjacent token $\hat{t}$ at the peak[3] in the LS may significantly influence and potentially improve the sensitivity of t. This operation actually smoothes the sensitivity peak of $\hat{t}$ in disguise and reduces the fluctuation of single data.

4.3 Dynamic Construction of Adjacency List

To better preserve token semantics while generating non-privacy text, we use the noisy semantic density to dynamically construct token adjacency lists.

4.3.1 Adjacency List Construction

We construct an adaptive-size adjacency list for each token t. Given a token $t \in V_t$, the adjacency list $C_r(t)$ consists of k_t tokens nearest to t considering the Euclidean distance in the embedding space: $C_r(t) = \{t_1, t_2, \cdots, t_{k_t}\}$, where k_t denotes the size of the token t's adjacency list. Note that $C_r(t)$ always contains at least token t itself.

4.3.2 Dynamic Adjacency List Using Noisy Semantic Density

To achieve fine-grained control over the adjacency list, we leverage the noisy semantic density obtained by DP-based semantic density information modeling to dynamically adjust each token's adjacency list size. The motivation stems from the limitations of existing studies, which either set a fixed adjacency list size (Yue et al., 2021; Chen et al., 2023) or apply a uniform noise distribution on all tokens (Tong et al., 2025) to determine the

[2]When noise is adjusted based on a token's LS, high sensitivity leads to larger noise amplitudes. If an attacker detects these changes, they could infer the token's local characteristics, potentially exposing privacy. For instance, density information may reveal the token's location in the embedding space.

[3]The occurrence of a peak means that the LS of a token is significantly higher than that of other adjacent tokens.

range of the adjacency list. However, for tokens with higher density (i.e. lower semantic sensitivity), enforcing the same strict privacy protection may result in unnecessary semantic loss. Therefore, to preserve the token's semantic information as much as possible, we aim to adjust the size of the adjacency list based on the token's sensitivity.

Specifically, we dynamically determine the size of the adjacency list of a token based on its density information. The process consists of two steps:

Step 1: Density Normalization. We apply a Min-Max normalization (Henderi et al., 2021) to the noisy density information $F(t)$ of token t, ensuring that the normalized value $\hat{F}(t)$ falls within $[0, 1]$, thereby adjusting the adjacency list size on a unified scale. F_{min} and F_{max} represent the minimum and maximum values of the density information for all tokens in the embedding space.

$$\hat{F}(t) = \frac{F(t) - F_{min}}{F_{max} - F_{min}}, \tag{7}$$

Min-max normalization linearly scales data, preserving its relative proportions. It retains the original distribution shape and statistical properties.

Step 2: Dynamic Scaling of Adjacency Lists. We use the normalized density information $\hat{F}(t)$ to scale the adjacency list size. With a default hyperparameter K as the maximum size, we obtain the adjacency list size k_t for token t as:

$$k_t = \max\left(1, \left\lfloor (1 - \hat{F}(t))K \right\rfloor\right). \tag{8}$$

Eq. 8 ensures that when $\hat{F}(t)$ is close to 0 (low density), k_t approaches K, creating a larger adjacency list; and when $\hat{F}(t)$ is close to 1 (high density), k_t approaches 1, resulting in a smaller adjacency list.

According to Eq. 8, the size of a token's adjacency list is inversely proportional to its noisy density information, enabling dynamic adjustment based on semantic density. Specifically, tokens with higher density have lower sensitivity (See §4.2 for analysis), resulting in a smaller adjacency list where the included tokens are semantically closer to the target token. This increases the likelihood of sampling closer tokens, effectively preserving the target token's semantic information. In contrast, tokens with lower density have higher sensitivity, resulting in a larger adjacency list that includes more tokens farther in semantic distance from the target token, thereby enhancing privacy protection.

4.4 Private Token Replacement via Similarity

For each sensitive token, we replace it with a perturbed non-sensitive token sampled from its adjacency list under DP protection. To achieve this, we design a replacement mechanism that integrates the exponential mechanism (McSherry and Talwar, 2007), ensuring the LDP guarantee while accounting for semantic relevance. We introduce similarity-based scoring to determine the probability of selecting a replacement token from the adjacency list.

Similarity-based Score. We design a scoring function $u(\cdot)$ for the replacement mechanism $M(\cdot)$. The goal is to assign higher scores to candidate tokens that exhibit greater semantic similarity to the target token, thereby increasing their probability of being sampled. Thus, we use the negative Euclidean distance and normalize it to the range $[0, 1]$. Specifically, for a token $t \in V_t$ and its candidate token $\hat{t} \in C_r(t)$, we first compute the Euclidean distance $d(t, \hat{t})$ to measure their semantic distance and define the scoring function as: $u(t, \hat{t}) = 1 - \frac{d(t,\hat{t})}{d_{max}}$, where d_{max} represents the semantic distance between token t and the farthest token t_{k_t} in its adjacency list as: $d_{\text{max}} = d(t, t_{k_t})$. Since $d(t, \hat{t}) \leq d_{max}$, it follows that $0 \leq \frac{d(t,\hat{t})}{d_{max}} \leq 1$. Consequently, we can deduce: $0 \leq u(t, \hat{t}) \leq 1, \Delta u = 1$.

Replacement Mechanism. Given the privacy budget parameter ε_r of the replacement module, for the input token $t \in V_t$, the probability (McSherry and Talwar, 2007) of the replacement mechanism $M(\cdot)$ outputting the candidate token $\hat{t} \in Cr(t)$ is:

$$\begin{aligned} Pr[M(t) = \hat{t}] &= \text{softmax}(\frac{\varepsilon_r \cdot u(t, \hat{t})}{2\Delta u}) \\ &= \frac{\exp(\frac{\varepsilon_r \cdot u(t,\hat{t})}{2\Delta u})}{\sum_{t_i \in C_r(t)} \exp(\frac{\varepsilon_r \cdot u(t,t_i)}{2\Delta u})} \end{aligned} \tag{9}$$

The replacement mechanism leverages the scoring function $u(t, \hat{t})$ to prioritize candidate tokens with higher semantic similarity in the adjacency list obtained in (§4.3), ensuring that tokens with closer tokens have a greater probability of being sampled. At the same time, the intensity of privacy protection can be flexibly controlled by adjusting the privacy budget ε_r. A higher privacy budget leads the mechanism to favor candidate tokens closer in semantics to the target token, while a lower budget increases randomness to strengthen privacy protection. We prove that the replacement mechanism satisfies ε_r-DP, with the detailed proof provided in the APP. A.

Method	IMDb						20 Newsgroups						PubMedQA					
	MAUVE			Coherence			MAUVE			Coherence			MAUVE			Coherence		
GPT-4	0.258			0.599			0.228			0.601			0.315			0.737		
Vicuna-7B	0.094			0.023			0.180			0.406			0.230			0.609		
	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$
FBDD	0.049	0.074	0.062	0.169	0.169	0.172	0.056	0.040	0.043	0.159	0.156	0.157	0.092	0.096	0.078	0.352	0.351	0.352
SANTEXT+	0.205	0.228	0.236	0.403	0.463	0.550	0.115	0.102	0.135	0.373	0.418	0.494	0.219	0.230	0.238	0.595	0.676	0.726
CUSTEXT+	0.225	0.252	0.197	0.588	0.580	0.550	0.153	0.171	0.152	0.557	0.562	0.562	0.183	0.224	0.219	0.693	0.698	0.703
RANTEXT	0.038	0.047	0.054	0.113	0.125	0.128	0.030	0.040	0.047	0.095	0.125	0.132	0.010	0.010	0.010	0.127	0.142	0.151
DYNTEXT	**0.241**	**0.254**	**0.242**	**0.589**	**0.590**	**0.590**	**0.183**	**0.180**	**0.158**	**0.578**	**0.579**	**0.579**	**0.271**	**0.289**	**0.341**	**0.727**	**0.728**	**0.732**

Table 1: Comparing the performance of all methods on open text generation tasks with different privacy budgets ($\varepsilon = 1, 2, 3$) on three datasets, evaluated using MAUVE and Coherence metrics. The best results are highlighted in bold. Our improvements are significant under the t-test with $p < 0.05$ (See details in App. E).

Method	IMDb						20 Newsgroups						PubMed QA					
	MAUVE			Coherence			MAUVE			Coherence			MAUVE			Coherence		
	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$
w/o smooth	0.241	0.248	0.242	0.589	0.589	0.585	0.158	0.124	0.157	0.295	0.294	0.295	0.269	0.287	0.299	0.726	0.726	0.729
w/o dynamic adj. list	0.235	0.248	0.240	0.586	0.587	0.589	0.156	0.139	0.155	0.294	0.294	0.292	0.246	0.287	0.254	0.726	0.728	0.727
w/o replacement	0.241	0.247	0.241	0.582	0.587	0.589	0.172	0.164	0.125	0.293	0.293	0.292	0.264	0.258	0.325	0.722	0.724	0.722
DYNTEXT	**0.241**	**0.254**	**0.242**	**0.589**	**0.590**	**0.590**	**0.183**	**0.180**	**0.158**	**0.578**	**0.579**	**0.579**	**0.271**	**0.289**	**0.341**	**0.727**	**0.728**	**0.732**

Table 2: Ablation results on DYNTEXT. *w/o* indicates that we remove a specific module or an approach from our full model. The best results are highlighted in bold.

5 Experiments

5.1 Experimental Settings

Datasets. For open-ended text generation tasks, we use three widely-used NLP datasets: IMDb, 20 Newsgroups, and PubMedQA (details in App. B).

Baselines. We use two non-DP methods as references: `GPT-4`, continues the original private text using GPT-4 without privacy protection. `Vicuna-7b`, continues the original private text using the local model Vicuna-7b (Chiang et al., 2023). We use four types of DP-based sanitization mechanisms to obtain the sanitized text, followed by text generation with GPT-4: `FBDD` (Feyisetan et al., 2020) adds noise to token embeddings and replaces the token with the token closest to the noisy embedding. `SANTEXT+` (Yue et al., 2021) applies the exponential mechanism to replace each token with a semantically similar one from the embedding space. `CUSTEXT+` (Chen et al., 2023) uses a fixed set of adjacent candidates and the exponential mechanism for replacement. `RANTEXT` (Tong et al., 2025) applies Laplace noise (Kotz et al., 2012) to introduce randomness into the non-sensitive token list and uses the exponential mechanism for replacement.

Metrics. Following (Tong et al., 2025), we evaluate the quality of the generated text with (see App. C for details): 1) `MAUVE` (Pillutla et al., 2021); 2) `Coherence`.

Details of implementation in App.D.

5.2 Overall Performance

Tab. 1 compares the continued text quality performance of all baselines across three benchmark datasets under different privacy budgets. Across all datasets, `DYNTEXT` consistently outperforms DP-based baselines in both MAUVE and Coherence, demonstrating superior text quality even under low privacy budgets. Specifically, (1) `GPT-4` typically represents the upper bound of performance, as it directly accesses the original private text. The quality of its generated text generally surpasses that of the local model `Vicuna`. (2) Despite the DP perturbation applied to the prompts, `DYNTEXT` generates text that closely approximates the quality of `GPT-4`. (3) In the PubmedQA dataset, focused on the medical privacy domain, `DYNTEXT` performs exceptionally well, achieving significant improvements over other baseline methods. This demonstrates that `DYNTEXT` excels in the privacy domain as well.

5.3 Ablation Study

Tab. 2 presents the ablation studies of `DYNTEXT`. The ablation results show that the full `DYNTEXT` consistently outperforms all other configurations, validating the effectiveness of each module. (1) `w/o smooth` uses GS instead of token-level smooth sensitivity (§ 4.2). The performance drops significantly on the 20 Newsgroups, indicating that using GS when there is abnormal data may introduce excessive noise, leading to poor performance. (2) `w/o dynamic adj. list` uses a fixed adjacency list of

Method	IMDb			20 Newsgroups			PubMedQA		
	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$
SANTEXT+	0.97143	0.97143	0.97143	0.01136	0.04735	0.18939	0.02667	0.11000	0.25556
CUSTEXT+	0.39778	0.38778	0.37333	0.31439	0.32955	0.36237	0.27333	0.26333	0.27111
RANTEXT	0.00243	0.01160	0.02439	0.00000	0.00192	0.00637	0.00000	0.00333	0.00222
DYNTEXT	**0.00008**	**0.00008**	**0.00007**	**0.00000**	**0.00010**	**0.00009**	**0.00000**	**0.00001**	**0.00002**

Table 3: Comparing the attack success rates (r_{ats}) of input inference attacks under different methods with different privacy budgets ($\varepsilon = 1, 2, 3$) on three datasets. Bold text denotes the best attack resistance.

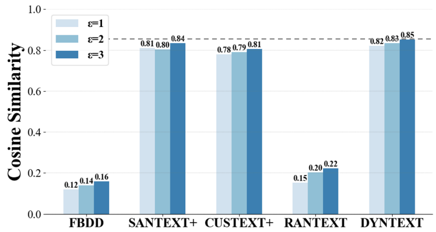


Figure 2: The cosine similarity between the replacement token and the original token in GloVe embedding obtained by different baselines with DP budgets.

size $\frac{2}{K}$ instead of dynamically adjusting the adjacency list size based density information (§ 4.3). The performance is significantly reduced, highlighting the effectiveness of the dynamic adjacency list in preserving semantics. (3) `w/o replacement` adds noise directly to the original token embedding, then finds the token closest to the noisy embedding in the dynamic adjacency list to replace the original token, instead of using the replacement mechanism (§ 4.4). The decline in results confirms that the replacement mechanism effectively samples semantically closer tokens while ensuring DP.

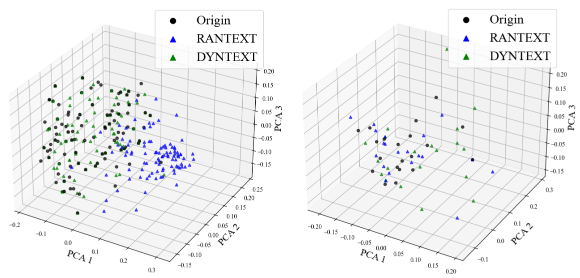
5.4 Analysis Study of Anti-attack

To evaluate the anti-attack capability of each method under different privacy budgets, we conduct input inference attack (Yue et al., 2021) experiments on three datasets and compute the attack success rate r_{ats}. In this attack, the adversary uses a pre-trained BERT model to recover the original private text from the perturbed text by masking and predicting each token. The attack is successful if the prediction matches the original token. The results in Tab. 3 show that DYNTEXT outperforms other baselines in privacy protection against input inference attacks, with r_{ats} approaching 0. Moreover, DYNTEXT maintains high stability as the privacy budget increases, unlike other baselines that rise significantly. This demonstrates DYNTEXT's robust and stable privacy protection capabilities.

5.5 Analysis Study of Token Similarity

To reflect the semantic loss caused by replacing sensitive tokens among different methods, we compare the similarity between the replacement tokens obtained by each method and the original token. Specifically, we measure the cosine similarity (Xia et al., 2015) between the original token and its replacement in the GloVe embedding (Pennington et al., 2014). As shown in Fig. 2: (1) For the same privacy budget ε, DYNTEXT achieves the highest cosine similarity, indicating minimal semantic loss. (2) As ε decreases, all methods show a decline in similarity, reflecting higher semantic loss with stronger privacy protection. (3) FBDD and Rantext show notably low cosine similarity, indicating that methods introduce significant semantic deviation.

5.6 Distribution in Different Density Areas



(a) High-density area. (b) Low-density area.

Figure 3: The original token distribution and the replacement token distribution of DYNTEXT and RANTEXT samples in different density areas.

We plot the token distribution of `Origin` and the sampling distributions of RANTEXT and DYNTEXT in different density areas. First, we reduce the high-dimensional space to three dimensions. Using Eq.4, we extract tokens from both high- and low-density areas and randomly sample some as original tokens. Then, we apply the sanitization mechanism to generate replacement tokens. From Fig.3, we observe: (1) In high-density (low-sensitivity) areas, DYNTEXT closely resembles `Origin`, preserving semantics well; while in low-density (high-sensitivity) areas, semantic deviation increases, enhancing privacy. (2) RANTEXT matches DYNTEXT in low-density areas but diverges in high-density areas, suggesting that RANTEXT applies the same privacy strategy to all tokens, leading to unnecessary semantic loss.

6 Conclusion

In summary, we proposed DYNTEXT to sanitize text for privacy-preserving LLM inference. DYNTEXT extracts token density using semantic-based density information modeling under DP; then dynamically constructs the adjacency list of each token based on the density information to adaptively adjust the protection level; finally, samples non-private tokens from the list through a replacement mechanism to replace sensitive tokens. Experiments show that DYNTEXT achieves SOTA performance in balancing the privacy-utility trade-off.

7 Limitations

In our study, several limitations warrant attention. Firstly, the current method has been exclusively validated within the context of single-language text continuation tasks. Considering that state-of-the-art models for other tasks, such as multilingual processing, machine translation, or text summarization, often incorporate complex components, substantial further research is necessary to adapt our model for these applications. In future work, we intend to extend DYNTEXT to new domains beyond text generation, including optimization for these intricate components, to enhance its versatility and performance across diverse scenarios.

Secondly, due to the current method's reliance on internal semantic information, it has not fully leveraged external knowledge bases, contextual data, or external retrieval mechanisms to augment semantic understanding. This limitation may result in inadequate identification and protection of sensitive information in complex scenarios, a prevalent challenge in this field. To address this issue, we plan to explore the integration of multi-source information into the privacy protection mechanism, aiming to further balance the trade-off between semantic retention and privacy safeguarding.

8 Ethical Considerations

We have rigorously proven through theoretical analysis that our method DYNTEXT satisfies DP guarantees and has demonstrated strong empirical security through adversarial attack experiments. However, residual theoretical risks of malicious exploitation still exist, particularly when processing sensitive medical or legal documents. Despite our experiments indicating nearly zero successful attacks, real-world adversaries may utilize unforeseen attack vectors. Consequently, for high-stakes applications such as healthcare or legal advice, we recommend augmenting our method with human reviews to ensure that outputs adhere to ethical and safety standards. We propose that users consider our method as a robust initial defense mechanism, complementing it with additional security measures to establish a comprehensive protection system. Future research will focus on further enhancements to mitigate these residual risks.

Acknowledgements

This work is supported by the following foundations: the National Natural Science Foundation of China (NSFC) under Grant No.62025208, No. 62421002, No.62306330, the National Science Foundation for Distinguished Young Scholars under Grant No. under Grant No. 62325604, No. 62125604, and the Young Elite Scientist Sponsorship Program by CAST under Grant No.YESS20230367.

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.

M. S. Bartlett. 1937. Properties of sufficiency and statistical tests. *Proc. R. Soc. Lond. A*, 160(901):268–282.

Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J Su. 2020. Deep learning with gaussian differential privacy. In *Harvard Data Science Review*, volume 2020. NIH Public Access.

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. 2023. A customized text sanitization mechanism with differential privacy. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5747–5758.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan

Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6.

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. 2006. Differential privacy. In *International colloquium on automata, languages, and programming*, pages 1–12. 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. 2024. Privacy preserving prompt engineering: A survey. *arXiv preprint arXiv:2404.06001*.

Tao Fan, Yan Kang, Weijing Chen, Hanlin Gu, Yuanfeng Song, Lixin Fan, Kai Chen, and Qiang Yang. 2024. Pdss: A privacy-preserving framework for step-by-step distillation of large language models. *arXiv preprint arXiv:2406.12403*.

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.

Oluwaseyi Feyisetan, Tom Diethe, and Thomas Drake. 2019. Leveraging hierarchical representations for preserving privacy and utility in text. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 210–219.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Henderi Henderi, Tri Wahyuningsih, and Efana Rahwanto. 2021. Comparison of min-max normalization and z-score normalization in the k-nearest neighbor (knn) algorithm to test the accuracy of types of breast cancer. *International Journal of Informatics and Information Systems*, 4(1):13–20.

Ken Huang, Fan Zhang, Yale Li, Sean Wright, Vasan Kidambi, and Vishwas Manral. 2023. Security and privacy concerns in chatgpt. In *Beyond AI: ChatGPT, Web3, and the Business Landscape of Tomorrow*, pages 297–328. Springer.

Bertrand Iooss and Paul Lemaître. 2015. A review on global sensitivity analysis methods. *Uncertainty management in simulation-optimization of complex systems: algorithms and applications*, pages 101–122.

Sanxiu Jiao, Jintao Meng, Yue Zhao, Kui Cheng, et al. 2024. Efficient dp-fl: Efficient differential privacy federated learning based on early stopping mechanism. *Computer Systems Science & Engineering*, 48(1).

Samuel Kotz, Tomasz Kozubowski, and Krzystof Podgorski. 2012. *The Laplace distribution and generalizations: a revisit with applications to communications, economics, engineering, and finance*. Springer Science & Business Media.

Alexey Kurakin, Natalia Ponomareva, Umar Syed, Liam MacDermed, and Andreas Terzis. 2023. Harnessing large-language models to generate private synthetic text. *arXiv preprint arXiv:2306.01684*.

K.) Lagler, K(Lagler, M.) Schindelegger, M(Schindelegger, J.) Bohm, J(Boehm, H.) Krasna, H(Krasna, and T.) Nilsson, T(Nilsson. 2013. Gpt2: Empirical slant delay model for radio space geodetic techniques. *GEOPHYSICAL RESEARCH LETTERS*.

Dongkyu Lee, Zhiliang Tian, Yingxiu Zhao, Ka Chun Cheung, and Nevin Zhang. 2022. Hard gate knowledge distillation-leverage calibration for robust and reliable language model. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9793–9803.

Xianzhi Li, Ran Zmigrod, Zhiqiang Ma, Xiaomo Liu, and Xiaodan Zhu. 2024. Fine-tuning language models with differential privacy through adaptive noise allocation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8368–8375.

Yansong Li, Zhixing Tan, and Yang Liu. 2025. Privacy-preserving prompt tuning for large language model services. *Preprint*, arXiv:2305.06212.

Lingjuan Lyu, Xuanli He, and Yitong Li. 2020. Differentially private representation for nlp: Formal guarantee and an empirical study on privacy and fairness. *arXiv preprint arXiv:2010.01285*.

Peihua Mai, Ran Yan, Zhe Huang, Youjia Yang, and Yan Pang. 2024. Split-and-denoise: Protect large language model inference with local differential privacy. In *Forty-first International Conference on Machine Learning*.

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.

Dung Nguyen, Mahantesh Halappanavar, Venkatesh Srinivasan, and Anil Vullikanti. 2024. Faster approximate subgraph counts with privacy. *Advances in Neural Information Processing Systems*, 36.

AI Open. 2023. Gpt-4 is openai's most advanced system. *Producing Safer and More Useful Responses.[Online].[cited 2024 May 9]*.

OpenAI. 2023a. Gpt-4 is openai's most advanced system and more useful responses. https://openai.com/gpt-4.

OpenAI. 2023b. New and improved embedding model. https://openai.com/blog/new-and-improved-embedding-model.

OpenAI. 2023c. Tokeniser for use with openai's models. https://github.com/openai/tiktoken.

Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. 2016. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers.

Anastasia Pustozerova, Jan Baumbach, and Rudolf Mayer. 2023. Differentially private federated learning: Privacy and utility analysis of output perturbation and dp-sgd. In *2023 IEEE International Conference on Big Data (BigData)*, pages 5549–5558.

Chen Qu, Weize Kong, Liu Yang, Mingyang Zhang, Michael Bendersky, and Marc Najork. 2021. Natural language understanding with privacy-preserving bert. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 1488–1497.

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

Yiping Song, Juhua Zhang, Zhiliang Tian, Yuxin Yang, Minlie Huang, and Dongsheng Li. 2024. Llm-based privacy data augmentation guided by knowledge distillation with a distribution tutor for medical text classification. *arXiv preprint arXiv:2402.16515*.

Pierre Tholoniat, Huseyin A Inan, Janardhan Kulkarni, and Robert Sim. 2024. Differentially private training of mixture of experts models. *arXiv preprint arXiv:2402.07334*.

Zhiliang Tian, Yingxiu Zhao, Ziyue Huang, Yu-Xiang Wang, Nevin L Zhang, and He He. 2022. Seqpate: Differentially private text generation via knowledge distillation. *Advances in Neural Information Processing Systems*, 35:11117–11130.

Meng Tong, Kejiang Chen, Jie Zhang, Yuang Qi, Weiming Zhang, Nenghai Yu, Tianwei Zhang, and Zhikun Zhang. 2025. Inferdpt: Privacy-preserving inference for black-box large language models. *IEEE Transactions on Dependable and Secure Computing*.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2019. Neural text generation with unlikelihood training. *Statistics*.

Matthew Wicker, Philip Sosnin, Adrianna Janik, Mark N Müller, Adrian Weller, and Calvin Tsay. 2024. Certificates of differential privacy and unlearning for gradient-based training. *arXiv preprint arXiv:2406.13433*.

Peipei Xia, Li Zhang, and Fanzhang Li. 2015. Learning similarity with cosine similarity ensemble. *Information sciences*, 307:39–52.

Jin Xu, Xiaojiang Liu, Jianhao Yan, Deng Cai, Huayang Li, and Jian Li. 2022. Learning to break the loop: Analyzing and mitigating repetitions for neural text generation.

Yadollah Yaghoobzadeh and Hinrich Schütze. 2016. Intrinsic subspace evaluation of word embedding representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 236–246.

Tao Yang and Xuebin Ma. 2024. Adadp-cfl: Cluster federated learning with adaptive clipping threshold differential privacy. In *2024 IEEE/ACM 32nd International Symposium on Quality of Service (IWQoS)*, pages 1–10. IEEE.

Tianyu Yang, Xiaodan Zhu, and Iryna Gurevych. 2024. Robust utility-preserving text anonymization based on large language models. *arXiv preprint arXiv:2407.11770*.

Zi Yin and Yuanyuan Shen. 2018. On the dimensionality of word embedding. *Advances in neural information processing systems*, 31.

Haoxiang Yuan, Xiaochen Yuan, Xiuli Bi, Weisheng Li, Guoyin Wang, and Bin Xiao. 2024. Pp-ldg: A medical privacy-preserving labeled data generation framework. In *2024 IEEE International Conference on Medical Artificial Intelligence (MedAI)*, pages 651–658.

Xiang Yue, Minxin Du, Tianhao Wang, Yaliang Li, Huan Sun, and Sherman SM Chow. 2021. Differential privacy for text analytics via natural text sanitization. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3853–3866. Association for Computational Linguistics (ACL).

Xiang Yue, Huseyin Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. 2023. Synthetic text generation with differential privacy: A simple and practical recipe. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1321–1342.

Haoqi Zheng, Qihuang Zhong, Liang Ding, Zhiliang Tian, Xin Niu, Changjian Wang, Dongsheng Li, and Dacheng Tao. 2023. Self-evolution learning for mixup: Enhance data augmentation on few-shot text classification tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8964–8974.

A ε_r-LDP Proof for the Replacement Mechanism

We need to prove that, given a privacy parameter $\varepsilon_r \geq 0$, for any two adjacent input tokens $t, t' \in V_t$ and output token $\hat{t} \in C_r(t) \wedge C_r(t')$, their probability ratio satisfies:

$$\frac{Pr[M(t) = \hat{t}]}{Pr[M(t') = \hat{t}]} \leq e^{\varepsilon_r}. \tag{10}$$

According to the probability formula Eq. 9 of the replacement mechanism, we expand the probability ratio:

$$\frac{Pr[M(t) = \hat{t}]}{Pr[M(t') = \hat{t}]} = \frac{\frac{\exp\left(\frac{\varepsilon_r \cdot u(t,\hat{t})}{2\Delta u}\right)}{\sum_{t_i \in C_r(t)} \exp\left(\frac{\varepsilon_r \cdot u(t,t_i)}{2\Delta u}\right)}}{\frac{\exp\left(\frac{\varepsilon_r \cdot u(t',\hat{t})}{2\Delta u}\right)}{\sum_{t_i \in C_r(t')} \exp\left(\frac{\varepsilon_r \cdot u(t',t_i)}{2\Delta u}\right)}}. \tag{11}$$

Because of $0 \leq u(t, \hat{t}) \leq 1, 0 \leq u(t', \hat{t}) \leq 1$ and $\Delta u = 1$, it can be further deduced that:

$$\begin{aligned} \frac{\exp\left(\frac{\varepsilon_r \cdot u(t,\hat{t})}{2\Delta u}\right)}{\exp\left(\frac{\varepsilon_r \cdot u(t',\hat{t})}{2\Delta u}\right)} &= \exp\left(\frac{\varepsilon_r}{2\Delta u}(u(t,\hat{t}) - u(t',\hat{t}))\right) \\ &\leq \exp\left(\frac{\varepsilon_r}{2}\right). \end{aligned} \tag{12}$$

We use the maximum-minimum ratio inequality to analyze the change in the denominator. Assumptions: (1) The smallest softmax normalization term in $C_r(t)$ corresponds to $\exp\left(\frac{\varepsilon_r \cdot u_{\min}(t)}{2}\right)$. (2) The largest softmax normalization term in $C_r(t')$ corresponds to $\exp\left(\frac{\varepsilon_r \cdot u_{\max}(t')}{2}\right)$. Therefore:

$$\begin{aligned} &\sum_{t_i \in C_r(t')} \exp\left(\frac{\varepsilon_r \cdot u(t', t_i)}{2}\right) \\ \leq &|C_r(t')| \cdot \exp\left(\frac{\varepsilon_r \cdot u_{\max}(t')}{2}\right), \end{aligned} \tag{13}$$

$$\sum_{t_i \in C_r(t)} \exp\left(\frac{\varepsilon_r \cdot u(t, t_i)}{2}\right) \geq \exp\left(\frac{\varepsilon_r \cdot u_{\min}(t)}{2}\right). \tag{14}$$

Thereby, it can be further deduced that:

$$\begin{aligned} &\frac{\sum_{t_i \in C_r(t')} \exp\left(\frac{\varepsilon_r \cdot u(t',t_i)}{2}\right)}{\sum_{t_i \in C_r(t)} \exp\left(\frac{\varepsilon_r \cdot u(t,t_i)}{2}\right)} \\ \leq &|C_r(t')| \cdot \exp\left(\frac{\varepsilon_r(u_{\max}(t') - u_{\min}(t))}{2}\right) \\ \leq &|C_r(t')| e^{\frac{\varepsilon_r}{2}}. \end{aligned} \tag{15}$$

By combining the changes in both the numerator and denominator, we obtain:

$$\frac{Pr[M(t) = \hat{t}]}{Pr[M(t') = \hat{t}]} \leq e^{\frac{\varepsilon_r}{2}} \cdot |C_r(t')| e^{\frac{\varepsilon_r}{2}} = |C_r(t')| e^{\varepsilon_r}. \tag{16}$$

Since in DYNTEXT, the size of the adjacency list $|C_r(t')|$ is a finite constant (at most K), the replacement mechanism satisfies ε_r-LDP. It can be proved:

$$\frac{Pr[M(t) = \hat{t}]}{Pr[M(t') = \hat{t}]} \leq e^{\varepsilon_r}. \tag{17}$$

So the replacement mechanism satisfies ε_r-DP.

B Details of Datasets

For open-ended text generation tasks, we employ three benchmark corpora comprising distinct scales and domains: (a) The IMDb dataset[4] (3,000 samples) provides movie review texts for binary sentiment analysis; (b) 20 Newsgroups[5] contains 1,766 documents across 20 thematic categories for multi-class news classification and (c) PubMedQA[6] (1,000 expert-annotated instances) supports biomedical question answering using research abstracts.

C Details of Metrics

Following previous works of open-ended text generation (Welleck et al., 2019; Xu et al., 2022; Tong et al., 2025), we use the first 50 tokens of the articles referred to as the raw document *Doc*, which requires privacy protection. We use the continuation writing of *Doc*, referred to as *Gen*, which consists of 100 tokens. Tokens are counted by the tokenization scheme of GPT-2 (Lagler et al., 2013). Following (Tong et al., 2025), we use two metrics

[4]https://huggingface.co/datasets/shubnandi/imdb_small

[5]https://huggingface.co/datasets/aihpi/20_newsgroups_demo

[6]https://huggingface.co/datasets/knowledgator/PubmedQA

Dataset	MAUNE			Coherence		
	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$
IMDb	1.78e-10	1.24e-02	8.08e-22	1.62e-37	8.22e-06	6.09e-19
20 Newsgroups	1.09e-16	1.79e-04	6.31e-06	4.61e-13	3.93e-10	1.28e-12
PubMedQA	1.65e-20	1.42e-24	1.31e-31	1.46e-35	4.95e-26	6.23e-03

Table 4: Statistical significance test results (p-values) across privacy budgets ε for **MAUNE** and **Coherence** metrics. All p-values < 0.05 confirm significant improvements over baselines.

to evaluate the quality of the generated text in the open-ended generation task:

1) **MAUVE** (Pillutla et al., 2021): It is used to assess the similarity between text generated by a language model and human-authored target continuation text.

2) **Coherence**: It calculates the cosine similarity between the text and the continuation.

$$COH(Doc, Gen) = \frac{\mathrm{SimCSE}(Doc) \cdot \mathrm{SimCSE}(Gen)}{|\mathrm{SimCSE}(Doc)| \cdot |\mathrm{SimCSE}(Gen)|} \tag{18}$$

where $\mathrm{SimCSE}(x) \in \mathbb{R}^d$ denotes the sentence embedding vector of x generated by the SimCSE model (Gao et al., 2021).

D Details of Implementation

The total privacy budget of DYNTEXT is $\varepsilon = \varepsilon_d + \varepsilon_r$. The privacy budget parameter ε_d defaults to 0.5. We set δ to 1×10^{-6} by default. Following Custext, we default K to 20. For black-box inference, we use GPT-4 (OpenAI, 2023a) to generate continuation text with the temperature parameter set to 0.5. Correspondingly, the token vocabulary V_t of GPT-4 is cl100k_base (OpenAI, 2023c). For the embedding function $\phi(\cdot)$, we select text-embedding-ada-002 (OpenAI, 2023b), which utilizes the same token vocabulary cl100k_base with GPT-4.

E Significance Test Results

We conduct the t-test (Bartlett, 1937) to examine whether the improvements of our method are significant. The p values in Tab. 4 are all smaller than 0.05, demonstrating the significance of our improvements.