Differentially Private Natural Language Models: Recent Advances and Future Directions

Lijie Hu[1,4]**, Ivan Habernal**[2]**, Lei Shen**[3]**, and Di Wang**[1,4]
[1]CEMSE, King Abdullah University of Science and Technology
[2]Department of Computer Science, Paderborn University
[3]JD AI Research, Beijing, China [4]SDAIA-KAUST AI
{lijie.hu, di.wang}@kaust.edu.sa, ivan.habernal@uni-paderborn.de, shenlei20@jd.com

Abstract

Recent developments in deep learning have led to great success in various natural language processing (NLP) tasks. However, these applications may involve data that contain sensitive information. Therefore, how to achieve good performance while also protecting the privacy of sensitive data is a crucial challenge in NLP. To preserve privacy, Differential Privacy (DP), which can prevent reconstruction attacks and protect against potential side knowledge, is becoming a de facto technique for private data analysis. In recent years, NLP in DP models (DP-NLP) has been studied from different perspectives, which deserves a comprehensive review. In this paper, we provide the first systematic review of recent advances in DP deep learning models in NLP. In particular, we first discuss some differences and additional challenges of DP-NLP compared with the standard DP deep learning. Then, we investigate some existing work on DP-NLP and present its recent developments from three aspects: gradient perturbation based methods, embedding vector perturbation based methods, and ensemble model based methods. We also discuss some challenges and future directions.

1 Introduction

The recent advances in deep neural networks have led to significant success in various tasks in Natural Language Processing (NLP), such as sentiment analysis, question answering, information retrieval, and text generation. However, such applications always involve data that contains sensitive information. For example, a model of aid typing on a keyboard which trained from language data might contain sensitive information such as passwords, text messages, and search queries. Moreover, language data can also identify a speaker explicitly by name or implicitly, for example, via a rare or unique phrase. Thus, one often encountered challenge in NLP is how to handle this sensitive information. To overcome the challenge, privacy-preserving NLP has been intensively studied in recent years. One of the commonly used approaches is based on text anonymization (Pilán et al., 2022), which identifies sensitive attributes and then replaces these sensitive words with some other values. Another approach is injecting additional words into the original text without detecting sensitive entities in order to achieve text redaction (Sánchez and Batet, 2016). However, removing personally identifiable information or injecting additional words is often unsatisfactory, as it has been shown that an adversary can still infer an individual's membership in the dataset with high probability via the summary statistics on the datasets (Narayanan and Shmatikov, 2008). Moreover, recent studies claim that deep neural networks for NLP tasks often tend to memorize their training data, which makes them vulnerable to leaking information about training data (Shokri et al., 2017; Carlini et al., 2021, 2019). One way that takes into account the limitations of existing approaches by preventing individual re-identification and protecting against any potential data reconstruction and side-knowledge attacks is designing Differentially Private (DP) algorithms. DP (Dwork et al., 2006) provides provable protection against identification and is resilient to arbitrary auxiliary information that might be available to attackers. Thanks to its formal guarantees, DP has become a de facto standard tool for private statistical data analysis.

Although there are numerous studies on DP machine learning and DP deep learning, such as (Abadi et al., 2016; Bu et al., 2019; Yu et al., 2019; Xiang et al., 2023; Xiao et al., 2023; Hu et al., 2023a,b), most of them mainly focus on either the continuous tabular data or image data, and less attention has been paid to adapting variants of DP algorithms to the context of NLP and the text domain. On the other side, while there are several surveys on DP and its applications, such as (Ji et al.,

Findings of the Association for Computational Linguistics: EACL 2024, pages 478–499
March 17-22, 2024 ©2024 Association for Computational Linguistics

2014; Dankar and Emam, 2013; Xiong et al., 2020; Wang et al., 2020b; Desfontaines and Pejó, 2020), none of them study its applications to the NLP domain. Recently, Klymenko et al. (2022) gave a brief introduction to applications of DP in NLP, but the reviewed work is not exhaustive, and it lacks a technical and systematic view of DP-NLP. Thus, to fill in this gap, in this paper, we provide the first technical overview of the recent developments and challenges of DP in language models.

Specifically, we give a survey on the most recent 70[1] papers on deep learning based approaches for NLP tasks under DP constraints. First, we show some specificities of DP-NLP compared with the general deep learning with DP. Then we discuss current results from three perspectives via the ways of adding randomness to ensure DP: the first one is gradient perturbation based methods which includes DP-SGD and DP-Adam; the second one is embedding vector perturbation based methods which includes DP auto-encoder; the last one is ensemble model based methods which includes PATE. For each type of approach, we also consider its applications to different NLP tasks. Finally, we present some potential challenges and future directions.

Due to space limits, in Appendix C, we give a preliminary introduction to DP to readers who are unfamiliar with DP.

2 Specificities of NLP with DP

We first discuss some specificities for DP-NLP compared with the standard DP deep learning. Generally speaking, there are two aspects: one is privacy notations, and another is privacy levels.

2.1 Variants of DP Notions in NLP

Recall that DP ensures data analysts or adversaries will get almost the same information if we change any single data sample in the training data, i.e., it treats all records as sensitive. However, such an assumption is quite stringent. On the one side, unlike image data, for text data, it is more common that only several instead of all attributes need to be protected. For example, for the sentence "My cell phone number is 1234567890", only the last token with the actual cell phone number needs to be protected. On the other side, canonical DP requires that the log of the ratio between the distribution probabilities is always upper bounded by the privacy parameter ϵ for any pair of neighboring data. However, such a requirement is also quite restrictive. For example, for the sentence "I will arrive at 2:00 pm", we want the adversary not to distinguish it from the sentence "I will arrive at 4:00 pm". However, DP also can ensure the adversary cannot distinguish it from the sentence "I will arrive at 100:00 pm", which is meaningless. Thus, for language data, besides the canonical DP, it is also reasonable to study its relaxations for some specific scenarios. Actually, this is quite different from the existing work on DP deep learning, which mainly focuses on standard DP definitions. In the following, we will discuss some commonly used relaxations of DP for language models.

SDP. As we mentioned above, in some scenarios, the sensitive information in text data is sparse, and we only need to protect some sensitive attributes instead of the whole sentence. Based on this, Shi et al. (2021) propose a new privacy notion, namely selective differential privacy (SDP), to provide privacy guarantees on the sensitive portion of the data to improve model utility. From the definition aspect, the main difference between SDP and DP is the definition of neighboring datasets. Informally, in SDP, two datasets are adjacent if they differ in at least one sensitive attribute. However, it is hard to define such neighboring datasets directly as there are some correlations between sensitive and non-sensitive attributes, indicating that we can still infer information on sensitive attributes (Kifer and Machanavajjhala, 2011). To address the issue, Shi et al. (2021) leverage the Pufferfish framework in (Kifer and Machanavajjhala, 2014).

Metric DP. To relax the requirement that the log probability ratio is uniformly bounded by ϵ for all neighboring data pairs, Feyisetan et al. (2020) first adopt the Metric DP (or d_χ-privacy) to the problem of private embedding, which is proposed by (Chatzikokolakis et al., 2013) for location data originally. In particular, a Metric DP mechanism could report a token in a privacy-preserving manner while giving a higher probability to tokens that are close to the current token, and a negligible probability to tokens in a completely different part of the vocabulary, where we will use some distance function d to measure the distance between two tokens.

[1] Note that we did not cover all related works, see the Limitations and Future Directions sections for the works that are not included in this paper.

Definition 1. For a data domain (vocabulary) $\mathcal{X}$, a randomized algorithm $\mathcal{A} : \mathcal{X} \mapsto \mathcal{R}$ is called (ε, δ)-Metric DP with distance function d if for any $S, S' \in \mathcal{X}^l$ and $T \subseteq \mathcal{R}$ we have

$$\text{Pr}[\mathcal{A}(S) \in T] \leq e^{d(S,S')\varepsilon}\text{Pr}[\mathcal{A}(S') \in T] + \delta.$$

From the above definition, we can see the probability ratio of observing any particular output y given two possible inputs S and S' is bounded by $e^{\varepsilon d(S',S)}$ instead of e^{ϵ} in DP. Motivated by Metric DP and local DP, (Feyisetan et al., 2020) provides the Local Metric DP (LMDP) and uses it for private word embeddings (see Section 4 for details). Motivated by Utility-optimized LDP (ULDP) (Murakami and Kawamoto, 2019) rather than LDP, recently Yue et al. (2021) propose Utility-optimized Metric LDP (UMLDP). It exploits the fact that different inputs have different sensitivity levels to achieve higher utility. By assuming the input space, such as the set of tokens is split into sensitive and non-sensitive parts, UMLDP achieves a privacy guarantee equivalent to LDP for sensitive inputs.

2.2 Variants Levels of Privacy in NLP

When we consider using DP, the first question is what kind of information we aim to protect. In the previous studies on DP deep learning, we always wanted to protect the whole data sample. However, in the NLP domain, such one data sample could be either a word, a sentence, a paragraph, etc. If we ignore the concrete privacy level and directly apply the previous DP methods, we may have mediocre results. Thus, unlike the sample level privacy in DP deep learning, researchers in NLP consider different levels of privacy. Especially, they focus on the word level and sentence level, which aims to protect each word and sentence respectively (Meehan et al., 2022; Feyisetan et al., 2019).

In the federated learning setting, there is a central server and several users each of them has a local dataset, the sample level of DP may be insufficient. For example, in language modeling, each user may contribute many thousands of words to the training data, and each typed word makes its own contribution to the RNN's training objective. In this case, just protecting each word is unsatisfactory, and it is still possible to re-identify users. Thus, besides the sample level, we also have the user level of privacy, which aims to protect users' histories. After discussing some specificities of DP-NLP. In the following, we categorize its recent studies into three classes based on their methods to ensure DP: gradient perturbation based methods, embedding vector perturbation based methods, and ensemble model based methods. See Tab. 1 for an overview.

3 Gradient Perturbation Based Methods

Generally speaking, a gradient perturbation method is based on adding noises to gradients of the loss during training the network to ensure DP. As the baseline and canonical algorithm for this type of approach, Differentially Private Stochastic Gradient Descent (DP-SGD) (Abadi et al., 2016) is a DP version of SGD. Its main idea is to use the noisy and clipped subsampled gradient g^t to approximate the whole gradient $\nabla L(\theta^t, D)$. In fact, besides SGD, we can use this idea for any optimizer, such as Adam (Kingma and Ba, 2015), whose private version DP-Adam is proposed and applied in BERT by (Anil et al., 2021). In the past few years, there has been a long list of work on DP-SGD from different perspectives, such as the subsampling strategy, faster clipping procedures, private clipping parameter tuning, and the selection of batch size. In the following, we will only discuss the previous work on using DP-SGD-based methods for variants of NLP tasks. See Appendix A for an introduction to DP-SGD.

3.1 DP Pre-trained Models

Recent developments in NLP have led to successful applications in large-scale language models with the appearance of transformer (Devlin et al., 2019). It combines the contextual information into language models with a more powerful ability of representation. These models are called pre-trained models, which train word embedding in large corpora targeting various tasks and gain the knowledge for downstream tasks (Peters et al., 2018). In this section, we review some papers that focus on pre-trained NLP models under DP constraints.

The workflow of BERT (Devlin et al., 2019) is pre-training the unlabeled text using some large corpora first. Then, the downstream tasks first initialize the model using the same parameters and fine-tune the parameters according to different tasks. Despite the benefits of powerful representation ability given by the pre-training process, it also has privacy issues since the model would memorize sensitive information such as words or phrases.

In order to solve this privacy leakage issue, there are several studies on how to train BERT privately. Hoory et al. (2021) successfully trained a differentially private BERT model by modifying the WordPiece algorithm to satisfy DP, and conducted experiments on the problem of entity extraction tasks from medical text. They construct a tailored domain-specific DP-based trained vocabulary designed to generate a new domain-specific vocabulary while maintaining user privacy and then use the original DP-SGD in the training process. For the DP vocabulary part, they first construct a word histogram by dividing the text into a sequence of N-word tuples and then add Gaussian noise to the histogram to ensure (ϵ, δ)-DP. Finally, they clip the histogram with some threshold. For the training phase, they use the original DP-SGD to meet privacy guarantees. Besides, they also use the parallel training trick to make the training faster. Very recently, Yin and Habernal (2022) applied DP-BERT to the legal NLP domain. While DP-BERT can achieve good performance with privacy guarantees in language tasks. There are still two problems: a large gap between non-private accuracy and private accuracy, and computation inefficiency of clipping every sample gradient in DP-SGD. In order to mitigate these issues, Anil et al. (2021) later privatizes the Adam optimizer to improve the performance. Instead of adding noise and clipping every entry in every batch in DP-SGD, it selects a pre-defined number of samples randomly and sums the clipped gradients of these selected samples, then it updates average gradients with Gaussian noise adding the sum in each batch. Besides, it also uses an increasing batch size schedule instead of a fixed one. It finds that large batch size can improve accuracy, and the increasing batch size schedule can improve training efficiency. (Senge et al., 2022) recently studied five different typical NLP tasks with varying complexity using modern neural models based on BERT and XtremeDistil architectures. They showed that to achieve adequate performance, each task and privacy regime requires special treatment.

Besides BERT, Ponomareva et al. (2022) privately pre-train T5 (Raffel et al., 2020) via their proposed private tokenizer called DP-SentencePiece and DP-SGD. They show that DP-T5 does not suffer a large drop in pre-training utility, nor in training speed, and can still be fine-tuned to high accuracy on downstream tasks.

3.2 DP Fine-tuning

Besides training pre-trained models using DP algorithms, another direction is how to fine-tune pre-trained models privately. Here, the main difference is that we assume the pre-trained models, such as BERT have been trained with some public data, and our goal is to privately fine-tune targeting specific downstream tasks that involve sensitive data. It is noted that in this section, we also include some related work on training shallow neural networks in DP such as RNN or LSTM such as (Li et al., 2022; Amid et al., 2022) as these methods can be directly applied to DP fine-tuning.

In this topic, the first direction is to investigate different tasks in the DP model and to compare its performance compared to the non-private one for studying the utility-privacy trade-off. Yue et al. (2022) consider the task of synthetic text generation and show that simply fine-tuning a pre-trained GPT-2 with the vanilla DP-SGD enables the model to generate useful synthetic text. Mireshghallah et al. (2022) recently extended to generating latent semantic parses in the DP model and then generating utterances based on the parses. Carranza et al. (2023) use DP-SGD to fine-tune a publicly pre-trained LLM on a query generation task. The resulting model can generate private synthetic queries representative of the original queries which can be freely shared for downstream non-private recommendation training procedures. Very recently, Lee and Søgaard (2023) adopted the DP-SGD to the meeting summarization task and showed that DP can improve performance when evaluated on unseen meeting types. Aziz et al. (2022) use GPT-2 and DP-SGD based methods to generate synthetic EHR data which can de-identify sensitive information for clinical text. Wunderlich et al. (2021) study the hierarchical text classification task, and they use DP-SGD to Bag of Words (BoW), CNNs and Transformer-based architectures. They find that Transformer-based models achieve better performance than CNN-based models in large datasets, while CNN-based models are superior to Transformer-based models in small datasets.

The second direction is to reduce the huge memory cost of storing individual gradients and decrease the added noise, which suffers notorious dimensional dependence in DP-SGD. Specifically, the studies in this direction always propose a general method for DP-SGD and then perform the method for different NLP tasks. Yu et al. (2021) propose a variant of DP-SGD called the Reparametrized Gradient Perturbation (RGP) method. The framework of RGP parametrizes each weight matrix with two low-rank carrier matrices and a residual weight matrix, which will be used to approximate the original one. Such a way can reduce the memory cost for computing individual gradient matrices and can maintain the optimization process via forward/backward signals. Later, based on RGP, Yu et al. (2022) show that advanced parameter-efficient methods such as (Houlsby et al., 2019; Karimi Mahabadi et al., 2021) can lead to simpler and significantly improved algorithms for private fine-tuning. Instead of DP-SGD, Du and Mi (2021) propose a DP version of Forward-Propagation. Specifically, it clips representations followed by noise addition in the forward propagation stage.

Besides adapting the optimization method in vanilla DP-SGD, there are also some works on modifying the clipping operation or the fine-tuning method directly to save the memory cost. Li et al. (2021) propose a memory-saving technique that allows clipping in DP-SGD for fine-tuning to run without instantiating per-example gradients for any linear layer in the model. The technique enables private training Transformers with almost the same memory cost as non-private training at a modest run-time overhead. Dupuy et al. (2021) propose another variant of DP-SGD via micro-batch computations per GPU and noise decay and apply it to fine-tuning models. Specifically, they scale gradients in each micro-batch and set a decreasing noise multiplier with epoch. Then, they add scaled Gaussian noise to gradients. In this way, they can make the training faster and adapt it for GPU training. Bu et al. (2023) develop a novel Book-Keeping (BK) technique that implements existing DP optimizers, with a substantial improvement on the computational cost while also keeping almost the same accuracy as DP-SGD. Gupta et al. (2023) propose a novel language transformer finetuning strategy that introduces task-specific parameters in multiple transformer layers. They show that the method of combining RGP and their novel strategy is more suitable for low-resource applications. Bu et al. (2022) privatize the bias-term fine-tuning (BiTFiT) and show that DP-BiTFiT matches the state-of-the-art accuracy for DP algorithms and the efficiency of the standard BiTFiT (Zaken et al., 2022). Igamberdiev and Habernal (2022) apply DP-Adam in Graph Convolutional Networks to perform the private fine-tuning for text classification. Specifically, they first split the graph into disconnected sub-graphs and then add noise to gradients.

Rather than reducing the memory cost, there are some papers considering developing variants of the DP-SGD method to improve performance. For example, Xia et al. (2023) propose a per-sample adaptive clipping algorithm, which is a new perspective and orthogonal to dynamic adaptive noise and coordinate clipping methods. Behnia et al. (2022) use the Edgeworth accountant (Wang et al., 2022) to compute the amount of noise that is required to be added to the gradients in SGD to guarantee a certain privacy budget, which is lower than the original DP-SGD. Li et al. (2022); Amid et al. (2022) propose new private optimization methods under the setting where there are some public and non-sensitive data.

The last direction is to relax the definition of DP and propose new DP-SGD variants. Shi et al. (2021) tailor DP-SGD to SDP. Their method SDP-SGD first splits the text into the sensitive and non-sensitive parts, and applies normal SGD to the non-sensitive part while applying DP-SGD to the sensitive part respectively. Later, Shi et al. (2022) extend to large language models and propose a method, namely Just Fine-tune Twice to private fine-tuning with the guarantee of SDP.

3.3 Federated Learning Setting

In the previous parts, we reviewed the related work on DP pre-trained models and DP fine-tuning models. Note that all the previous work only considers the central DP setting where all the training data samples are already collected before training, indicating that these methods cannot be applied to the federated learning (FL) setting. Compared to central DP, there are fewer studies on DP Federated Learning for NLP. McMahan et al. (2018) apply DP-SGD in the FedAvg algorithm to protect user-level privacy for LSTM and RNN architectures in the federated learning setting. Specifically, they first sample users with some probability, and then add Gaussian noise to model updates of the sampled users on the server side. Based on this, Ramaswamy et al. (2020) develop the first consumer-scale next-word prediction model.

Rather than adopting DP-SGD, Kairouz et al. (2021) provides a new paradigm for DP-FL by using the Follow-The-Regularized-Leader (FTRL) algorithm, which achieves state-of-the-art performance, and it is recently improved by Choquette-Choo et al. (2022); Koloskova et al. (2023); Denisov et al. (2022); Agarwal et al. (2021).

It is notable that all the previous studies only consider shallow neural networks such as RNN and LSTM and do not consider the large language model. Until very recently, there have been some papers studying DP-FL fine-tuning. For example, Wang et al. (2023) consider the cross-device setting and use DP-FTRL to privately fine-tune. Moreover, they propose a distribution matching algorithm that leverages both private on-device LMs and public LLMs to select public records close to private data distribution. Xu et al. (2023) deploy DP-FL versions of Gboard Language Models (Hard et al., 2018) via DP-FTRL and quantile-based clip estimation method in Andrew et al. (2021).

4 Embedding Vector Perturbation Based Methods

Generally speaking, this type of approach considers privatizing the embedding vector for each token. Specifically, in this framework, the text data is first transformed into a vector (text representation) via some word embedding method such as Word2Vec (Mikolov et al., 2013) and BERT. Then we use some DP mechanism to privatize each representation and train NLP models based on these privatized text representations. Due to the post-processing property of DP, we can see the main strength of this approach is any further training on these private embeddings also preserves the DP property, while gradient perturbation based methods heavily rely on the network structure. We can see that the main step of this method is to design the best private text representation. Note that since we need to privatize each embedding representation separately, the whole algorithm could be considered as an LDP algorithm, and thus, it can also be used in the LDP setting. It is also notable that different studies may consider different notions and levels of privacy. In fact, most of the existing work considers the word level of privacy.

4.1 Vanilla DP

The most direct approach is to design private embedding mechanisms that satisfy the standard DP. Lyu et al. (2020b) first study this problem and they propose a framework. Specifically, firstly, for each word, the embedding module of such framework outputs a 1-dimensional real representation with length r, then it privatizes the vector via a variant of the Unary Encoding mechanism in (Wang et al., 2017). In order to remove the dependence of dimensionality in the Unary Encoding mechanism, they propose an Optimized Multiple Encoding, which embeds vectors with a certain fixed size. Their post-processing procedure was then improved by (Plant et al., 2021). In (Plant et al., 2021), it first gets the final layer representation of the pre-trained model for each token, then normalizes it with sequence and adds Laplacian noise, and finally trains this classifier with adversarial training. To further improve the fairness for the downstream tasks on private embedding, later Lyu et al. (2020a) propose to dropout perturbed embeddings to amplify privacy and a robust training algorithm that incorporates the noisy training representation in the training process to derive a robust target model, which also reduces model discrimination in most cases.

Krishna et al. (2021); Habernal (2021); Alnasser et al. (2021) also study privatizing word embeddings. However, instead of using the Unary Encoding mechanism or dropout, Krishna et al. (2021); Alnasser et al. (2021) propose ADePT, which is an auto-encoder-based DP algorithm. Let $\mathbf{u}$ be the input, an auto-encoder model consists of an encoder that returns a vector representation $\mathbf{r} = \text{Enc}(\mathbf{u})$ for the input $\mathbf{u}$, which is then passed into the decoder to construct an output $\mathbf{v} = \text{Dec}(\mathbf{r})$. In (Krishna et al., 2021), it first normalized the word embedded vector by some parameter C i.e., $w = \text{Enc(u)} \min\{1, \frac{C}{\|\text{Enc}(u)\|_2}\}$, then it adds Laplacian noise to the normalized vector w and get $\mathbf{r}$. Unfortunately, Habernal (2021) points out that ADePT is not differentially private by thorough theoretical proof. The problem of ADePT lies in the sensitivity calculation and could be remedied by adding calibrated noise or tighter bounded clipping norm. Later, Igamberdiev et al. (2022) provides the source code of DP Auto-Encoder methods to improve reproducibility. Recently, Maheshwari et al. (2022) proposed a method that combines differential privacy and adversarial training techniques to solve the privacy-fairness-accuracy trade-off in local DP. In their framework, first, the input text will be fed into encoders, then it will be normalized and privatized by using the Laplacian mechanism. Next, it will be fed into a normal classifier and adversarial training separately to combine a loss that contains normal classification loss and adversarial loss. They find that the model can improve privacy and fairness simultaneously. To further improve the performance, (Bollegala et al., 2023) propose a Neighbourhood-Aware Differential Privacy (NADP) mechanism considering the neighborhood of a word in a pre-trained static word embedding space to determine the minimal amount of noise required to guarantee a specified privacy level.

Besides the work on word-level privacy we mentioned above, recently, there have been some works studying sentence-level and token-level private embeddings. Meehan et al. (2022) propose a method, namely DeepCandidate, to achieve sentence-level privacy. They first put public and private sentences into a sentence encoder to get sentence embeddings. Then, they use a method, namely DeepCandidate, to choose the candidate sentence embeddings that are near to private embeddings. Finally, they use some DP mechanism to sample from the candidate embeddings for each private embedding. This method somehow solves the challenge of the sentence-level privacy problem by taking advantage of clustering in differential privacy. (Du et al., 2023b) consider sentence-level privacy for private fine-tuning and propose DP-Forward fine-tuning, which perturbs the forward pass embeddings of every user's (labeled) sequence. However, it is notable that they consider a variant of LDP called sequence local DP. Chen et al. (2023) propose a novel Customized Text (CusText) sanitization mechanism that provides more advanced privacy protection at the token level.

4.2 Metric DP

In Metric DP for text data, each sample of the input can be represented as a string x with at most l words, thus, the data universe will be W^{ℓ} where W is a dictionary. Also we assume that there is a word embedding model $\phi : W \mapsto \mathbb{R}^n$ and its associated distance $d(x, x') = \sum_{i=1}^{l} \|\phi(w_i) - \phi(w_i')\|_2$, where $x = w_1 w_2 \cdots w_l$ and $x' = w_1' w_2' \cdots w_l'$ are two samples. Thus, the goal is to design a mechanism for each $\phi(w_i)$ with the guarantee of Metric DP. Since we aim to randomize each $\phi(w_i)$ for each sample. The whole algorithm is also suitable for local metric DP with word-level privacy.

Feyisetan et al. (2020) first study this problem. Generally speaking, their mechanism consists of two steps. The first step is perturbation, we add some noise N to text vector $\phi(w_i)$ to ensure ε-LDP, where N has the density probability function $p_N(z) \propto \exp(-\varepsilon\|z\|_2)$. The main issue of this approach is that after the perturbation, $\hat{\phi}_i$ may be inconsistent with the word embedding. That is, there may not exist a word u such that $u = \hat{\phi}_i$. Thus, to address this issue, we need to project the perturbed vector into the embedding space. That is the second step. Feyisetan et al. (2020) show that the algorithm is ε-local Metric DP.

Note that the method was later improved from different aspects. For example, Xu et al. (2020) reconsider the problem setting and they observe that the distance used in (Feyisetan et al., 2020) is the Euclidean norm $d(x, x') = \sum_{i=1}^{l} \|\phi(w_i) - \phi(w_i')\|_2$, which cannot describe the similarity between two words in the embedding space. To address the issue, they propose to use the Mahalanobis Norm and modify the algorithm by using the Mahalanobis mechanism, which can improve performance. To further improve the utility in the projection step, Xu et al. (2021b) further propose the Vickrey mechanism in case the first nearest neighbors are the original input or some rare words need large-scale noise to perturb and hard to find the corresponding words. In order to solve this problem, they use a hyperparameter in their algorithm to adjust the selection of the first and second nearest neighbors (words). To further allow a smaller range of nearby words to be considered than the multivariate Laplace mechanism, (Xu et al., 2021a; Carvalho et al., 2021b) propose an improved perturbation method via the Truncated Gumbel Noise. To further address the high dimensional issue, Feyisetan and Kasiviswanathan (2021) uses the random projection for the original text representation to a lower dimensional space and then projects back to the original space after adding random noise to preserve DP. Besides, Feyisetan et al. (2019) define the hyperbolic embeddings and use the Metropolis-Hastings (MH) algorithm to sample from hyperbolic distribution. However, it is remarkable that if we consider the LDP setting, then all the previous methods need to send real numbers to the server, which has a high communication cost. To address the issue, Carvalho et al. (2021a) proposes to use the binary randomized response mechanism by using binary embedding vectors. Recently, Tang et al. (2020) consider the case where different words may have different levels of privacy. They first divide the words into two types, and then add corresponding noise according to different levels of privacy. Imola et al. (2022) recently proposed an optimal Meric DP mechanism for finite vocabulary, they then provided an algorithm that could quickly calculate the mechanism. Finally, they applied it to private word embedding. Instead of developing new private mechanisms, there are also some studies on improving the embedding process. The previous metric DP mechanisms are expected to fall short of finding substitutes for words with ambiguous meanings. To address these ambiguous words, Arnold et al. (2023a) provide a sense embedding and incorporate a sense disambiguation step prior to noise injection. Arnold et al. (2023b) account for the common semantic context issue that appeared in the previous private embedding mechanisms. They incorporate grammatical categories into the privatization step in the form of a constraint to the candidate selection and show that selecting a substitution with matching grammatical properties amplifies the performance in downstream tasks. Qu et al. (2021) recently points out that (Lyu et al., 2020a) does not address privacy issues in the training phase since the server needs users' raw data to fine-tune. Moreover, its method has a high computational cost due to the heavy encoder workload on the user side. Thus, Qu et al. (2021) improve it and consider the federated setting where users send their privatized samples via some local metric DP mechanism to the server, and the server conducts privacy-constrained fine-tuning methods. Moreover, besides the text-to-text privatization given in (Feyisetan et al., 2020) and the sequence private representation proposed by Lyu et al. (2020a), Qu et al. (2021) proposed new token-level privatization and text-to-text privatization methods. In the token representation privatization method, they add random noise using metric DP to token embedding and send it to the server. They add noise to the embedded token and output the closest neighbor token in the embedding space.

Instead of the local Metric DP, Yue et al. (2021) consider UMLDP and propose SANTEXT and SANTEXT+ algorithms for text sanitization tasks. Specifically, they divide all the text into a sensitive token set $\mathcal{V}_S$ and a remaining token set $\mathcal{V}_N$. Then $\mathcal{V}_S$ and $\mathcal{V}_N$ will use a privacy budget of ϵ and ϵ_0 respectively via the composition theorem in LDP. After deriving token vectors, SANTEXT samples new tokens via local Metric DP with Euclidean distance. Compared with SANTEXT, SANTEXT+ samples new tokens when the original tokens are in sensitive set $\mathcal{V}_S$. They apply it to BERT pre-training and fine-tuning models.

While there are many studies on the benefits of private embedding with word-level privacy. There are also some shortcomings to such notion of privacy, as mentioned by (Mattern et al., 2022) recently. For example, in the previous private word embedding methods, we need to assume the length of the string for each sample is the same. Moreover, since we consider the word level of privacy, the total privacy budget will grow linearly with the length of the sample. To mitigate some shortcomings, Mattern et al. (2022) propose an alternative text anonymization method based on fine-tuning large language models for paraphrasing. To ensure DP, they adopt the exponential mechanism to sample from the softmax distribution. They apply their method in fine-tuning models with GPT-2.

Recently, Du et al. (2023a) studied sentence-level private embedding in local metric DP. Borrowing the wisdom of normalizing sentence embedding for robustness, they impose a consistency constraint on their sanitization. They propose two instantiations from the Euclidean and angular distances. The first one utilizes the Purkayastha mechanism (Weggenmann and Kerschbaum, 2021), and the other is upgraded from the generalized planar Laplace mechanism with post-processing.

Very recently, besides pre-training and fine-tuning, private word embedding has also been used in the task of prompt tuning for Large Language Models. The goal of private prompt tuning is to protect the privacy of examples demonstrated in the prompt. Specifically, Li et al. (2023) leverages the above private embedding methods to ensure local metric DP. To mitigate the performance degradation when imposing privacy protection, they propose a privatized token reconstruction task motivated by the recent findings that the masked language modeling objective can learn separable deep representations. Then, the objective of privatized token reconstruction is to recover the original content of a privatized special token sequence from LLM representations.

5 Challenges and Future Directions

DP for LLMs. Dealing with large-scale text data and training LLMs like GPT-4 are tough tasks in deep learning with DP. Due to the high dimensionality of embedding vectors, even adding small noise can have a significant influence on the training speed and performance of models. It is more severe for DP-SGD-based methods, which need high memory costs, and their per-example clipping procedure is time-consuming. These methods will be inefficient when they are applied to large language models. Thus, how to reduce the memory cost and accelerate the training or fine-tuning of DP-SGD become core concerns in gradient perturbation-based methods. Although there is some work in this direction, from Table 1 we can see most of the current studies are only for BERT, GPT-2, and T5, and there is still a gap in accuracy between private and non-private models and these methods still need catastrophic cost of memory compared with the non-private ones. Moreover, it is well known that we need a heavy workload on hyperparameter-tuning for large-scale models in the non-private case. From the privacy view, each try-on hyperparameter-tuning will cost an additional privacy budget, which makes our final private model cost a large privacy budget. Thus, how to efficiently and privately tune the hyperparameters in large models is challenging.

Besides the central setting, from Table 1, we can also see that DP training and In-context learning in the federated learning setting is still lacking in studies. Moreover, even for DP fine-tuning, we can see the current studies only focused on small models such as LaMDA, and there is still no study on private fine-tuning for LLMs in the federated learning setting.

Sentence-level Private Embedding As we mentioned, in embedding vector perturbation-based methods, the core problem is how to derive a private embedding that can avoid information leakage while also having good performance for downstream tasks. These methods use variants of distances to extract the relationship between words in the embedding space and use different noises to obfuscate sensitive tokens. Besides, some work focuses on how to use these private embeddings in specific settings like the generation of synthetic private data, federated learning, and fine-tuning models. However, these papers only focus on word-level privacy and do not consider sentence-level privacy which is more practical in the NLP scenario. For example, even if we replace some sensitive words (like name) using private embedding methods in a question-answering system, we can still easily infer that person from some sentences. In total, we should not only consider the privacy issue of each word but also consider how to hide sentence structures and syntax in sentences. Thus, designing sentence-level private embeddings is an important but difficult problem in private language models.

Private Inference. It is notable that in this paper, we mainly discussed how to privately train and release a language model without leaking information about training data. However, in some scenarios (such as Machine Learning as a Service), we only want to use the model for inference instead of releasing the model. Thus, for these scenarios, we only need to perform inference tasks based on our trained model, while we do not want to leak information about training data. From the DP side, such private inference corresponds to the DP prediction algorithm, which is proposed by (Dwork and Feldman, 2018). Compared with private training, DP inference for text data is still far from well-understood, and there are only few studies on it (Ginart et al., 2022; Majmudar et al., 2022).

Limitations

First, in this paper, we mainly focused on the deep learning-based models for NLP tasks in the differential privacy model. Actually, there are also some studies on classical statistical models or approaches for NLP in DP, such as topic modeling (Park et al., 2016; Zhao et al., 2021; Huang and Chen, 2021) and n-gram extraction (Kim et al., 2021). Secondly, due to the space limit, we did not discuss all the related work for DP-SGD, and we only focused on the work that uses DP-SGD to NLP-related tasks. Thirdly, while we tried our best to discuss all the existing work on deep learning-based methods for DP-NLP, we have to say that we may have missed some related work. Moreover, since we aim to classify all the current work into three categories based on their methods of adding randomness, there is still some work that does not belong to these three classes, such as (Bo et al., 2021; Weggenmann et al., 2022). To make our paper be consistent, we did not mention these works here. Fourthly, although DP can provide rigorous guarantees of privacy-preserving, it has also been shown that DP machine learning models can cause fairness issues. For example, they always have a disparate impact on model accuracy (Bagdasaryan et al., 2019). Finally, it is notable that in this paper, we did not discuss the narrow assumptions made by differential privacy, and the broadness of natural language and of privacy as a social norm. More details can be found in (Brown et al., 2022).

Acknowledgments

Di Wang and Lijie Hu are supported in part by the baseline funding BAS/1/1689-01-01, funding from the CRG grand URF/1/4663-01-01, FCC/1/1976-49-01 from CBRC, and funding from the AI Initiative REI/1/4811-10-01 of King Abdullah University of Science and Technology (KAUST). Di Wang and Lijie Hu are also supported by the funding of the SDAIA-KAUST Center of Excellence in Data Science and Artificial Intelligence (SDAIA-KAUST AI).

References

Martín Abadi, Andy Chu, Ian J. 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, Vienna, Austria, October 24-28, 2016*, pages 308–318. ACM.

Naman Agarwal, Peter Kairouz, and Ziyu Liu. 2021. The skellam mechanism for differentially private federated learning. *Advances in Neural Information Processing Systems*, 34:5052–5064.

Walaa Alnasser, Ghazaleh Beigi, and Huan Liu. 2021. Privacy preserving text representation learning using BERT. In *Social, Cultural, and Behavioral Modeling - 14th International Conference, SBP-BRiMS 2021, Virtual Event, July 6-9, 2021, Proceedings*, volume 12720 of *Lecture Notes in Computer Science*, pages 91–100. Springer.

Ehsan Amid, Arun Ganesh, Rajiv Mathews, Swaroop Ramaswamy, Shuang Song, Thomas Steinke, Vinith M Suriyakumar, Om Thakkar, and Abhradeep Thakurta. 2022. Public data-assisted mirror descent for private model training. In *International Conference on Machine Learning*, pages 517–535. PMLR.

Galen Andrew, Om Thakkar, Brendan McMahan, and Swaroop Ramaswamy. 2021. Differentially private learning with adaptive clipping. *Advances in Neural Information Processing Systems*, 34:17455–17466.

Rohan Anil, Badih Ghazi, Vineet Gupta, Ravi Kumar, and Pasin Manurangsi. 2021. Large-scale differentially private BERT. *CoRR*, abs/2108.01624.

Stefan Arnold, Dilara Yesilbas, and Sven Weinzierl. 2023a. Driving context into text-to-text privatization. *CoRR*, abs/2306.01457.

Stefan Arnold, Dilara Yesilbas, and Sven Weinzierl. 2023b. Guiding text-to-text privatization by syntax. *CoRR*, abs/2306.01471.

Shahab Asoodeh, Jiachun Liao, Flávio P. Calmon, Oliver Kosut, and Lalitha Sankar. 2021. Three variants of differential privacy: Lossless conversion and applications. *IEEE J. Sel. Areas Inf. Theory*, 2(1):208–222.

Md Momin Al Aziz, Tanbir Ahmed, Tasnia Faequa, Xiaoqian Jiang, Yiyu Yao, and Noman Mohammed. 2022. Differentially private medical texts generation using generative neural networks. *ACM Trans. Comput. Heal.*, 3(1):5:1–5:27.

Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. 2019. Differential privacy has disparate impact on model accuracy. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 15453–15462.

Borja Balle, Gilles Barthe, and Marco Gaboardi. 2018. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6280–6290.

Borja Balle, Gilles Barthe, Marco Gaboardi, and Joseph Geumlek. 2019. Privacy amplification by mixing and diffusion mechanisms. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13277–13287.

Borja Balle, Peter Kairouz, Brendan McMahan, Om Dipakbhai Thakkar, and Abhradeep Thakurta. 2020. Privacy amplification via random check-ins. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Rouzbeh Behnia, Mohammadreza Reza Ebrahimi, Jason Pacheco, and Balaji Padmanabhan. 2022. Ew-tune: A framework for privately fine-tuning large language models with differential privacy. In *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 560–566. IEEE.

Haohan Bo, Steven H. H. Ding, Benjamin C. M. Fung, and Farkhund Iqbal. 2021. ER-AE: Differentially private text generation for authorship anonymization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3997–4007, Online. Association for Computational Linguistics.

Danushka Bollegala, Shuichi Otake, Tomoya Machide, and Ken-ichi Kawarabayashi. 2023. A neighbourhood-aware differential privacy mechanism for static word embeddings. *CoRR*, abs/2309.10551.

Hannah Brown, Katherine Lee, Fatemehsadat Mireshghallah, Reza Shokri, and Florian Tramèr. 2022. What does it mean for a language model to preserve privacy? In *FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022*, pages 2280–2292. ACM.

Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J. Su. 2019. Deep learning with gaussian differential privacy. *CoRR*, abs/1911.11607.

Zhiqi Bu, Yu-Xiang Wang, Sheng Zha, and George Karypis. 2022. Differentially private bias-term only fine-tuning of foundation models. *arXiv preprint arXiv:2210.00036*.

Zhiqi Bu, Yu-Xiang Wang, Sheng Zha, and George Karypis. 2023. Differentially private optimization on large model at small cost. In *International Conference on Machine Learning*, pages 3192–3218. PMLR.

Mark Bun, Cynthia Dwork, Guy N. Rothblum, and Thomas Steinke. 2018. Composable and versatile privacy via truncated CDP. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 74–86. ACM.

Mark Bun and Thomas Steinke. 2016. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part I*, volume 9985 of *Lecture Notes in Computer Science*, pages 635–658.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*, pages 267–284. USENIX Association.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, pages 2633–2650. USENIX Association.

Aldo Gael Carranza, Rezsa Farahani, Natalia Ponomareva, Alex Kurakin, Matthew Jagielski, and Milad Nasr. 2023. Privacy-preserving recommender systems with synthetic query generation using differentially private large language models. *arXiv preprint arXiv:2305.05973*.

Ricardo Silva Carvalho, Theodore Vasiloudis, and Oluwaseyi Feyisetan. 2021a. BRR: preserving privacy of text data efficiently on device. *CoRR*, abs/2107.07923.

Ricardo Silva Carvalho, Theodore Vasiloudis, and Oluwaseyi Feyisetan. 2021b. TEM: high utility metric differential privacy on text. *CoRR*, abs/2107.07928.

Konstantinos Chatzikokolakis, Miguel E. Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi. 2013. Broadening the scope of differential privacy using metrics. In *Privacy Enhancing Technologies - 13th International Symposium, PETS 2013, Bloomington, IN, USA, July 10-12, 2013. Proceedings*, volume 7981 of *Lecture Notes in Computer Science*, pages 82–102. Springer.

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, Toronto, Canada, July 9-14, 2023*, pages 5747–5758. Association for Computational Linguistics.

Albert Cheu, Adam D. Smith, Jonathan R. Ullman, David Zeber, and Maxim Zhilyaev. 2019. Distributed differential privacy via shuffling. In *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I*, volume 11476 of *Lecture Notes in Computer Science*, pages 375–403. Springer.

Christopher A Choquette-Choo, H Brendan McMahan, Keith Rush, and Abhradeep Thakurta. 2022. Multi-epoch matrix factorization mechanisms for private machine learning. *arXiv preprint arXiv:2211.06530*.

Edwige Cyffers and Aurélien Bellet. 2022. Privacy amplification by decentralization. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event*, volume 151 of *Proceedings of Machine Learning Research*, pages 5334–5353. PMLR.

Fida Kamal Dankar and Khaled El Emam. 2013. Practicing differential privacy in health care: A review. *Trans. Data Priv.*, 6(1):35–67.

Sergey Denisov, H Brendan McMahan, John Rush, Adam Smith, and Abhradeep Guha Thakurta. 2022. Improved differential privacy for sgd via optimal private linear operators on adaptive streams. *Advances in Neural Information Processing Systems*, 35:5910–5924.

Damien Desfontaines and Balázs Pejó. 2020. Sok: Differential privacies. *Proceedings on Privacy Enhancing Technologies*, 2020(2):288–313.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA,* *June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Jinshuo Dong, Aaron Roth, and Weijie J. Su. 2022. Gaussian differential privacy. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 84(1):3–37.

Jian Du and Haitao Mi. 2021. Dp-fp: Differentially private forward propagation for large models. *arXiv preprint arXiv:2112.14430*.

Minxin Du, Xiang Yue, Sherman S. M. Chow, and Huan Sun. 2023a. Sanitizing sentence embeddings (and labels) for local differential privacy. In *Proceedings of the ACM Web Conference 2023*, WWW '23, page 2349–2359, New York, NY, USA. Association for Computing Machinery.

Minxin Du, Xiang Yue, Sherman S. M. Chow, Tianhao Wang, Chenyu Huang, and Huan Sun. 2023b. Dp-forward: Fine-tuning and inference on language models with differential privacy in forward pass. *CoRR*, abs/2309.06746.

Haonan Duan, Adam Dziedzic, Nicolas Papernot, and Franziska Boenisch. 2023. Flocks of stochastic parrots: Differentially private prompt learning for large language models. *CoRR*, abs/2305.15594.

Christophe Dupuy, Radhika Arava, Rahul Gupta, and Anna Rumshisky. 2021. An efficient DP-SGD mechanism for large scale NLP models. *CoRR*, abs/2107.14586.

Cynthia Dwork and Vitaly Feldman. 2018. Privacy-preserving prediction. In *Conference On Learning Theory, COLT 2018, Stockholm, Sweden, 6-9 July 2018*, volume 75 of *Proceedings of Machine Learning Research*, pages 1693–1702. PMLR.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. 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*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer.

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

Cynthia Dwork and Guy N. Rothblum. 2016. Concentrated differential privacy. *CoRR*, abs/1603.01887.

Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. 2010. Boosting and differential privacy. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 51–60. IEEE Computer Society.

Oluwaseyi Feyisetan, Borja Balle, Thomas Drake, and Tom Diethe. 2020. Privacy- and utility-preserving textual analysis via calibrated multivariate perturbations. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, pages 178–186. ACM.

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 2019, Beijing, China, November 8-11, 2019*, pages 210–219. IEEE.

Oluwaseyi Feyisetan and Shiva Kasiviswanathan. 2021. Private release of text embedding vectors. In *Proceedings of the First Workshop on Trustworthy Natural Language Processing*, pages 15–27.

Antonio Ginart, Laurens van der Maaten, James Zou, and Chuan Guo. 2022. Submix: Practical private prediction for large-scale language models. *CoRR*, abs/2201.00971.

Sivakanth Gopi, Yin Tat Lee, and Lukas Wutschitz. 2021. Numerical composition of differential privacy. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 11631–11642.

Umang Gupta, Aram Galstyan, and Greg Ver Steeg. 2023. Jointly reparametrized multi-layer adaptation for efficient and private tuning. *arXiv preprint arXiv:2305.19264*.

Ivan Habernal. 2021. When differential privacy meets NLP: The devil is in the detail. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1522–1528, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. 2018. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*.

Shlomo Hoory, Amir Feder, Avichai Tendler, Sofia Erell, Alon Peled-Cohen, Itay Laish, Hootan Nakhost, Uri Stemmer, Ayelet Benjamini, Avinatan Hassidim, and Yossi Matias. 2021. Learning and evaluating a differentially private pre-trained language model. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1178–1189, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Lijie Hu, Shuo Ni, Hanshen Xiao, and Di Wang. 2022. High dimensional differentially private stochastic optimization with heavy-tailed data. In *Proceedings of the 41st ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 227–236.

Lijie Hu, Zihang Xiang, Jiabin Liu, and Di Wang. 2023a. Nearly optimal rates of privacy-preserving sparse generalized eigenvalue problem. *IEEE Transactions on Knowledge and Data Engineering*.

Lijie Hu, Zihang Xiang, Jiabin Liu, and Di Wang. 2023b. Privacy-preserving sparse generalized eigenvalue problem. In *International Conference on Artificial Intelligence and Statistics*, pages 5052–5062. PMLR.

Tao Huang and Hong Chen. 2021. Improving privacy guarantee and efficiency of latent dirichlet allocation model training under differential privacy. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 143–152. Association for Computational Linguistics.

Timour Igamberdiev, Thomas Arnold, and Ivan Habernal. 2022. DP-Rewrite: Towards Reproducibility and Transparency in Differentially Private Text Rewriting. In *The 29th International Conference on Computational Linguistics*, pages 2927–2933, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Timour Igamberdiev and Ivan Habernal. 2022. Privacy-Preserving Graph Convolutional Networks for Text Classification. In *Proceedings of the Language Resources and Evaluation Conference*, pages 338–350, Marseille, France. European Language Resources Association.

Jacob Imola and Kamalika Chaudhuri. 2021. Privacy amplification via bernoulli sampling. *CoRR*, abs/2105.10594.

Jacob Imola, Shiva Prasad Kasiviswanathan, Stephen White, Abhinav Aggarwal, and Nathanael Teissier. 2022. Balancing utility and scalability in metric differential privacy. In *Uncertainty in Artificial Intelligence, Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence, UAI 2022, 1-5 August 2022, Eindhoven, The Netherlands*, volume 180 of *Proceedings of Machine Learning Research*, pages 885–894. PMLR.

Zhanglong Ji, Zachary Chase Lipton, and Charles Elkan. 2014. Differential privacy and machine learning: a survey and review. *CoRR*, abs/1412.7584.

Peter Kairouz, Brendan McMahan, Shuang Song, Om Thakkar, Abhradeep Thakurta, and Zheng Xu. 2021. Practical and private (deep) learning without sampling or shuffling. In *International Conference on Machine Learning*, pages 5213–5225. PMLR.

Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2015. The composition theorem for differential privacy. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1376–1385. JMLR.org.

Rabeeh Karimi Mahabadi, James Henderson, and Sebastian Ruder. 2021. Compacter: Efficient low-rank hypercomplex adapter layers. *Advances in Neural Information Processing Systems*, 34:1022–1035.

Daniel Kifer and Ashwin Machanavajjhala. 2011. No free lunch in data privacy. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, Athens, Greece, June 12-16, 2011*, pages 193–204. ACM.

Daniel Kifer and Ashwin Machanavajjhala. 2014. Pufferfish: A framework for mathematical privacy definitions. *ACM Trans. Database Syst.*, 39(1):3:1–3:36.

Kunho Kim, Sivakanth Gopi, Janardhan Kulkarni, and Sergey Yekhanin. 2021. Differentially private n-gram extraction. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 5102–5111.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Oleksandra Klymenko, Stephen Meisenbacher, and Florian Matthes. 2022. Differential privacy in natural language processing the story so far. In *Proceedings of the Fourth Workshop on Privacy in Natural Language Processing*, pages 1–11, Seattle, United States. Association for Computational Linguistics.

Anastasia Koloskova, Ryan McKenna, Zachary Charles, Keith Rush, and Brendan McMahan. 2023. Convergence of gradient descent with linearly correlated noise and applications to differentially private learning. *arXiv preprint arXiv:2302.01463*.

Satyapriya Krishna, Rahul Gupta, and Christophe Dupuy. 2021. ADePT: Auto-encoder based differentially private text transformation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2435–2439, Online. Association for Computational Linguistics.

Seolhwa Lee and Anders Søgaard. 2023. Private meeting summarization without performance loss. *arXiv preprint arXiv:2305.15894*.

Tian Li, Manzil Zaheer, Sashank Reddi, and Virginia Smith. 2022. Private adaptive optimization with side information. In *International Conference on Machine Learning*, pages 13086–13105. PMLR.

Xuechen Li, Florian Tramer, Percy Liang, and Tatsunori Hashimoto. 2021. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*.

Yansong Li, Zhixing Tan, and Yang Liu. 2023. Privacy-preserving prompt tuning for large language model services. *CoRR*, abs/2305.06212.

Lingjuan Lyu, Xuanli He, and Yitong Li. 2020a. Differentially private representation for NLP: Formal guarantee and an empirical study on privacy and fairness. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2355–2365, Online. Association for Computational Linguistics.

Lingjuan Lyu, Yitong Li, Xuanli He, and Tong Xiao. 2020b. Towards differentially private text representations. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 1813–1816. ACM.

Gaurav Maheshwari, Pascal Denis, Mikaela Keller, and Aurélien Bellet. 2022. Fair nlp models with differentially private text encoders. *arXiv preprint arXiv:2205.06135*.

Jimit Majmudar, Christophe Dupuy, Charith Peris, Sami Smaili, Rahul Gupta, and Richard S. Zemel. 2022. Differentially private decoding in large language models. *CoRR*, abs/2205.13621.

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

H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning differentially private recurrent language models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Casey Meehan, Khalil Mrini, and Kamalika Chaudhuri. 2022. Sentence-level privacy for document embeddings. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3367–3380, Dublin, Ireland. Association for Computational Linguistics.

Sebastian Meiser and Esfandiar Mohammadi. 2018. Tight on budget?: Tight bounds for r-fold approximate differential privacy. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, Toronto, ON, Canada, October 15-19, 2018*, pages 247–264. ACM.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Fatemehsadat Mireshghallah, Richard Shin, Yu Su, Tatsunori Hashimoto, and Jason Eisner. 2022. Privacy-preserving domain adaptation of semantic parsers. *arXiv preprint arXiv:2212.10520*.

Ilya Mironov. 2017. Rényi differential privacy. In *30th IEEE Computer Security Foundations Symposium, CSF 2017, Santa Barbara, CA, USA, August 21-25, 2017*, pages 263–275. IEEE Computer Society.

Ilya Mironov, Kunal Talwar, and Li Zhang. 2019. Rényi differential privacy of the sampled gaussian mechanism. *CoRR*, abs/1908.10530.

Takao Murakami and Yusuke Kawamoto. 2019. Utility-optimized local differential privacy mechanisms for distribution estimation. In *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*, pages 1877–1894. USENIX Association.

Jack Murtagh and Salil P. Vadhan. 2016. The complexity of computing the optimal composition of differential privacy. In *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I*, volume 9562 of *Lecture Notes in Computer Science*, pages 157–175. Springer.

Arvind Narayanan and Vitaly Shmatikov. 2008. Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (S&P 2008), 18-21 May 2008, Oakland, California, USA*, pages 111–125. IEEE Computer Society.

Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, and Kunal Talwar. 2016. Semi-supervised knowledge transfer for deep learning from private training data. In *International Conference on Learning Representations*.

Mijung Park, James R. Foulds, Kamalika Chaudhuri, and Max Welling. 2016. Private topic modeling. *CoRR*, abs/1609.04120.

Manas A. Pathak, Shantanu Rane, and Bhiksha Raj. 2010. Multiparty differential privacy via aggregation of locally trained classifiers. In *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, pages 1876–1884. Curran Associates, Inc.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. The text anonymization benchmark (TAB): A dedicated corpus and evaluation framework for text anonymization. *CoRR*, abs/2202.00443.

Richard Plant, Dimitra Gkatzia, and Valerio Giuffrida. 2021. CAPE: Context-aware private embeddings for private language learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7970–7978, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Chen Qu, Weize Kong, Liu Yang, Mingyang Zhang, Michael Bendersky, and Marc Najork. 2021. Natural language understanding with privacy-preserving BERT. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 1488–1497. ACM.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Swaroop Ramaswamy, Om Thakkar, Rajiv Mathews, Galen Andrew, H Brendan McMahan, and Françoise Beaufays. 2020. Training production language models without memorizing user data. *arXiv preprint arXiv:2009.10031*.

David Sánchez and Montserrat Batet. 2016. C-sanitized: A privacy model for document redaction and sanitization. *J. Assoc. Inf. Sci. Technol.*, 67(1):148–163.

Manuel Senge, Timour Igamberdiev, and Ivan Habernal. 2022. One size does not fit all: Investigating strategies for differentially-private learning across NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7340–7353, Abu Dhabi, UAE.

Weiyan Shi, Aiqi Cui, Evan Li, Ruoxi Jia, and Zhou Yu. 2021. Selective differential privacy for language modeling. *CoRR*, abs/2108.12944.

Weiyan Shi, Ryan Shea, Si Chen, Chiyuan Zhang, Ruoxi Jia, and Zhou Yu. 2022. Just fine-tune twice: Selective differential privacy for large language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6327–6340.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 3–18. IEEE Computer Society.

Jingye Tang, Tianqing Zhu, Ping Xiong, Yu Wang, and Wei Ren. 2020. Privacy and utility trade-off for textual analysis via calibrated multivariate perturbations. In *Network and System Security - 14th International Conference, NSS 2020, Melbourne, VIC, Australia, November 25-27, 2020, Proceedings*, volume 12570 of *Lecture Notes in Computer Science*, pages 342–353. Springer.

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. *CoRR*, abs/2309.11765.

Zhiliang Tian, Yingxiu Zhao, Ziyue Huang, Yu-Xiang Wang, Nevin L. Zhang, and He He. 2022. Seqpate: Differentially private text generation via knowledge distillation. In *NeurIPS*.

Boxin Wang, Yibo Jacky Zhang, Yuan Cao, Bo Li, H Brendan McMahan, Sewoong Oh, Zheng Xu, and Manzil Zaheer. 2023. Can public large language models help private cross-device federated learning? *arXiv preprint arXiv:2305.12132*.

Di Wang, Marco Gaboardi, and Jinhui Xu. 2018. Empirical risk minimization in non-interactive local differential privacy revisited. *Advances in Neural Information Processing Systems*, 31.

Di Wang, Hanshen Xiao, Srinivas Devadas, and Jinhui Xu. 2020a. On differentially private stochastic convex optimization with heavy-tailed data. In *International Conference on Machine Learning*, pages 10081–10091. PMLR.

Hua Wang, Sheng Gao, Huanyu Zhang, Milan Shen, and Weijie J Su. 2022. Analytical composition of differential privacy via the edgeworth accountant. *arXiv preprint arXiv:2206.04236*.

Teng Wang, Xuefeng Zhang, Jingyu Feng, and Xinyu Yang. 2020b. A comprehensive survey on local differential privacy toward data statistics and analysis. *Sensors*, 20(24).

Tianhao Wang, Jeremiah Blocki, Ninghui Li, and Somesh Jha. 2017. Locally differentially private protocols for frequency estimation. In *26th USENIX Security Symposium, USENIX Security 2017, Vancouver, BC, Canada, August 16-18, 2017*, pages 729–745. USENIX Association.

Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. 2020c. Subsampled rényi differential privacy and analytical moments accountant. *J. Priv. Confidentiality*, 10(2).

Benjamin Weggenmann and Florian Kerschbaum. 2021. Differential privacy for directional data. In *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021*, pages 1205–1222. ACM.

Benjamin Weggenmann, Valentin Rublack, Michael Andrejczuk, Justus Mattern, and Florian Kerschbaum. 2022. DP-VAE: human-readable text anonymization for online reviews with differentially private variational autoencoders. In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, pages 721–731. ACM.

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

Dominik Wunderlich, Daniel Bernau, Francesco Aldà, Javier Parra-Arnau, and Thorsten Strufe. 2021. On the privacy-utility trade-off in differentially private hierarchical text classification. *CoRR*, abs/2103.02895.

Tianyu Xia, Shuheng Shen, Su Yao, Xinyi Fu, Ke Xu, Xiaolong Xu, and Xing Fu. 2023. Differentially private learning with per-sample adaptive clipping. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37(9), pages 10444–10452.

Zihang Xiang, Tianhao Wang, Wanyu Lin, and Di Wang. 2023. Practical differentially private and byzantine-resilient federated learning. *Proceedings of the ACM on Management of Data*, 1(2):1–26.

Hanshen Xiao, Zihang Xiang, Di Wang, and Srinivas Devadas. 2023. A theory to instruct differentially-private learning via clipping bias reduction. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 2170–2189. IEEE Computer Society.

Xingxing Xiong, Shubo Liu, Dan Li, Zhaohui Cai, and Xiaoguang Niu. 2020. A comprehensive survey on local differential privacy. *Secur. Commun. Networks*, 2020:8829523:1–8829523:29.

Nan Xu, Oluwaseyi Feyisetan, Abhinav Aggarwal, Zekun Xu, and Nathanael Teissier. 2021a. Density-aware differentially private textual perturbations using truncated gumbel noise. In *Proceedings of the Thirty-Fourth International Florida Artificial Intelligence Research Society Conference, North Miami Beach, Florida, USA, May 17-19, 2021*.

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

Zekun Xu, Abhinav Aggarwal, Oluwaseyi Feyisetan, and Nathanael Teissier. 2021b. On a utilitarian approach to privacy preserving text generation. *arXiv preprint arXiv:2104.11838*.

Zekun Xu, Abhinav Aggarwal, Oluwaseyi Feyisetan, and Nathanael Teissier. 2021c. On a utilitarian approach to privacy preserving text generation. *CoRR*, abs/2104.11838.

Zheng Xu, Yanxiang Zhang, Galen Andrew, Christopher A Choquette-Choo, Peter Kairouz, H Brendan McMahan, Jesse Rosenstock, and Yuanbo Zhang. 2023. Federated learning of gboard language models with differential privacy. *arXiv preprint arXiv:2305.18465*.

Ying Yin and Ivan Habernal. 2022. Privacy-Preserving Models for Legal Natural Language Processing. In *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 172–183, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A. Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, Sergey Yekhanin, and Huishuai Zhang. 2022. Differentially private fine-tuning of language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. 2021. Large scale private learning via low-rank reparametrization. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12208–12218. PMLR.

Lei Yu, Ling Liu, Calton Pu, Mehmet Emre Gursoy, and Stacey Truex. 2019. Differentially private model publishing for deep learning. In *2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019*, pages 332–349. IEEE.

Xiang Yue, Minxin Du, Tianhao Wang, Yaliang Li, Huan Sun, and Sherman S. M. 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, Online. Association for Computational Linguistics.

Xiang Yue, Huseyin A Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Huan Sun, David Levitan, and Robert Sim. 2022. Synthetic text generation with differential privacy: A simple and practical recipe. *arXiv preprint arXiv:2210.14348*.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. 2022. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–9.

Fangyuan Zhao, Xuebin Ren, Shusen Yang, Qing Han, Peng Zhao, and Xinyu Yang. 2021. Latent dirichlet allocation model training with differential privacy. *IEEE Trans. Inf. Forensics Secur.*, 16:1290–1305.

Qinqing Zheng, Jinshuo Dong, Qi Long, and Weijie J. Su. 2020. Sharp composition bounds for gaussian differential privacy via edgeworth expansion. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 11420–11435. PMLR.

Yuqing Zhu and Yu-Xiang Wang. 2019. Poission subsampled rényi differential privacy. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7634–7642. PMLR.

A An Introduction to DP-SGD

Given a training data with n samples $D = \{x_i\}_{i=1}^n$, a loss function (such as cross-entropy loss) is defined to train the model, which takes the parameter $\theta \in \mathbb{R}^d$ of neural network and samples and outputs a real value:

$$L(\theta, D) = \sum_{i=1}^{n} \ell(\theta, x_i). \qquad (1)$$

The goal is to find the weights of the network that minimizes $L(\theta, D)$, *i.e.,* $\theta^* = \arg\min_\theta L(\theta, D)$. With additional constraint on DP, now we aim to design an $(\varepsilon, \delta)/\varepsilon$-DP algorithm $\mathcal{A}$ to make the private estimated parameter θ_{priv} close to θ^*.

Example: In Language Modeling (LM), we have a corpus $D = \{x_1, \cdots, x_n\}$ where each text sequence x_i consists of multiple tokens $x_i = (x_{i1}, \cdots, x_{im_i})$ with x_{ij} as the j-th token of x_i. The goal of LM is to train a neural network (e.g., RNN) parameterized by θ to learn the probability of the sequence $p_\theta(x)$, which can be represented as the following objective function

$$-\sum_{i=1}^{n}\sum_{j=1}^{m_i} \log p_\theta(x_{ij}|x_{i1}, \cdots, x_{i(j-1)}).$$

We first review the DP-SGD method (Abadi et al., 2016; Wang et al., 2018, 2020a; Hu et al., 2022). In the non-private case, to minimize the objective function (1), the most fundamental method is SGD, i.e., in the t-th iteration, we update the model as follows:

$$\theta^{t+1} = \theta^t - \eta \frac{1}{|B|} \sum_{x \in B} \nabla \ell(\theta^t, x),$$

where B is a subsampled batch of random examples, η is the learning rate and θ^t is the current parameter. DP-SGD modifies the SGD-based methods by adding Gaussian noise to perturb the (stochastic) gradient in each iteration of the training, *i.e,* during the t-th iteration DP-SGD will compute a noisy gradient as follows:

$$g^t = \frac{1}{|B|}(\sum_{x_i \in B} \hat{g}_i^t + \mathcal{N}\left(0, \sigma^2 C^2 I_d\right)), \qquad (2)$$

σ is noise multiplier, $\hat{g}_i^t$ is some vector computed from $\nabla \ell(\theta^t, x_i)$ and g^t is the (noisy) gradient used to update the model. The main reason here we use $\hat{g}_i^t$ instead of the original gradient vector is that we wish to make the term $\sum \hat{g}_i^t$ has bounded ℓ_2-sensitivity so that we can use the Gaussian mechanism to ensure DP. The most commonly used approach to get a $\hat{g}_i^t$ is clipping the gradient: $\hat{g}_i^t = \nabla \ell(\theta^t, x_i) \min\{1, \frac{C}{\|\nabla \ell(\theta^t, x_i)\|_2}\}$ *i.e.*, each gradient vector is clipped by a hyper-parameter $C > 0$.

Since the ℓ_2-sensitivity of $\sum \hat{g}_i^k$ is bounded by C, after the clipping, we can add Gaussian noise to ensure DP. As there are several iterations and in each iteration, we use some subsampling strategy, we can use the composition theorem and privacy amplification to compute the total privacy cost of DP-SGD. Equivalently, given a fixed privacy budget (ϵ, δ), number of iterations and subsampling strategy, one can get the minimal noise multiplier σ to ensure DP, see (Asoodeh et al., 2021; Gopi et al., 2021; Mironov et al., 2019; Wang et al., 2020c; Zheng et al., 2020; Zhu and Wang, 2019) for details.

B Ensemble Model Based Methods

Unlike gradient perturbation and private embedding based methods, the general idea of ensemble model based methods is first we divide the whole private data into several subsets, then we **non-privately** train a model for each private subset. To ensure privacy, for each time of inference or query, we will do a private aggregation for all models. Compared with the previous two types of approach, the main advantage of the ensemble model based method is the noise we add will be independent of the scale of the model or the dimension of the embedding space, indicating the noise is much smaller. However, the weakness is that here, we cannot release private embeddings or the private model, and each query or inference will cost a privacy budget. Generally speaking, based on

Method Type	**Publications**	**Scenarios**	**Definition**	**Model Architecture**	**DP Level**	**Downsteam Tasks**
Gradient Perturbation Based Methods	Hoory et al. (2021)	**Pre-trained**	DP	BERT	Sample-level	Entity-extraction
	Anil et al. (2021)			BERT	Sample-level	—
	Yin and Habernal (2022)			BERT	Sample-level	Classification, QA
	Senge et al. (2022)			BERT, XtremeDistil	Sample-level	Classification, NER, POS, QA
	Ponomareva et al. (2022)			T5	Sample-level	NLU
	Yu et al. (2022)	**Fine-tuning**	DP	RoBERT, GPT-2	Sample-level	NLG, NLU
	Yu et al. (2021)			BERT	Sample-level	Classification, NLU
	Dupuy et al. (2021)			BERT,BiLSTM	Sample-level	Classification, NER
	Li et al. (2021)			GPT-2, (Ro)BERT	Sample-level	Classification, NLG
	Lee and Søgaard (2023)			GPT-2, DialoGPT	Sample-level	Meeting Summarization
	Xia et al. (2023)			GPT-2, (Ro)BERT	Sample-level	Classification
	Behnia et al. (2022)			(Ro)BERT	Sample-level	NLU
	Bu et al. (2023)			GPT-2, (Ro)BERT	Sample-level	Classification
	Gupta et al. (2023)			(Ro)BERT	Sample-level	GLU
	Du and Mi (2021)			GPT-2, (Ro)BERT	Sample-level	Classification, NLG
	Bu et al. (2022)			(Ro)BERT	Sample-level	Classification, NLG
	Yue et al. (2022)			GPT-2	Sample-level	Synthetic Text Generation
	Mireshghallah et al. (2022)			GPT-2	Sample-level	Synthetic Text Generation
	Carranza et al. (2023)			T5	Sample-level	Query Generation
	Igamberdiev and Habernal (2022)			GPT-2	Sample-level	Classification
	Aziz et al. (2022)			GPT-2	Sample-level	Synthetic Text Generation
	Wunderlich et al. (2021)			BERT,CNN	Sample-level	Classification
	Li et al. (2022)			LSTM	Sample-level	Classification
	Amid et al. (2022)			LSTM	Sample-level	Classification
	Shi et al. (2021)		**SDP**	RNN	Sample-level	NLG, Dialog System
	Shi et al. (2022)		**SDP**	GPT-2, (Ro)BERT	Sample-level	NLG, NLU
	McMahan et al. (2018)	**Federated Learning**	**LDP**	LSTM, RNN	User-level	Prediction, Classification
	Ramaswamy et al. (2020)			LSTM	User-level	Prediction, Classification
	Kairouz et al. (2021)			LSTM	User-level, Sample-level	Prediction, Classification
	Choquette-Choo et al. (2022)			LSTM	User-level, Sample-level	Prediction
	Koloskova et al. (2023)			LSTM	User-level, Sample-level	Prediction
	Denisov et al. (2022)			LSTM	User-level, Sample-level	Prediction
	Agarwal et al. (2021)			LSTM	User-level, Sample-level	Prediction
	Wang et al. (2023)			LaMDA	User-level	Prediction
	Xu et al. (2023)			Gboard	User-level	Prediction
Embedding Vector Perturbation Based Methods	Lyu et al. (2020b)	Private Embedding	**LDP**	BERT	Word-level	Classification
	Lyu et al. (2020a)			BERT	Word-level	Classification
	Plant et al. (2021)			BERT	Word-level	Classification
	Krishna et al. (2021)			Auto-Encoder	Word-level	Classification
	Habernal (2021)			Auto-Encoder	Word-level	Classification
	Alnasser et al. (2021)			Auto-Encoder	Word-level	Classification
	Igamberdiev et al. (2022)			Auto-Encoder	Word-level	Classification
	Maheshwari et al. (2022)			Auto-Encoder	Word-level	Classification
	Bollegala et al. (2023)			GloVe	Word-level	Classification
	Chen et al. (2023)			GloVe, BERT	Token-level	Classification
	Du et al. (2023b)	Fine-tuning	Sequence LDP	BERT	Sentence-level	Classification, QA
	Meehan et al. (2022)	Private Embedding	DP	SBERT	**Sentence-level**	Classification
	Feyisetan et al. (2020)	Private Embedding	**LMDP**	GloVe, BiLSTM	Word-level	Classification, QA
	Xu et al. (2020)			GloVe	Word-level	Classification
	Xu et al. (2021c)			GloVe,FastText	Word-level	Classification
	Xu et al. (2021a)			GloVe, CNN	Word-level	Classification
	Carvalho et al. (2021b)			GloVe	Word-level	Classification
	Feyisetan and Kasiviswanathan (2021)			GloVe, FastText	Word-level	Classification
	Feyisetan et al. (2019)			GloVe	Word-level	Classification, Prediction
	Carvalho et al. (2021a)			GloVe, FastText	Word-level	Classification
	Tang et al. (2020)			GloVe	Word-level	Classification
	Imola et al. (2022)			GloVe, FastText	Word-level	Classification
	Arnold et al. (2023a)			GloVe	Word-level	Classification
	Arnold et al. (2023b)			GloVe	Word-level	Classification
	Qu et al. (2021)	Fine-tuning		BERT, BiLSTM	Token-level	Classification,NLU
	Du et al. (2023a)	Private Embedding		BERT	Sentence-level	Classification, QA
	Li et al. (2023)	Private Prompt Tuning		BERT, TA	Word-level	Classification, QA
	Yue et al. (2021)	Private Embedding	**UMLDP**	BERT, GloVe	Word-level	Classification,QA
Ensemble Model Based Methods	Duan et al. (2023)	**In-context Learning**		GPT-3	Sample-level	Classification
	Wu et al. (2023)			GPT-3	Sample-level	Classification, QA, Dialog Summarization
	Tang et al. (2023)			GPT-3	Sample-level	Classification, Information Extraction
	Tian et al. (2022)	**Fine-tuning**		GPT-2	Sample-level, User-level	Synthetic Text Generation

Table 1: An overview of studies for DP-NLP.

different private aggregations, there are two types of approaches: the PATE-based method, and the Sample-and-Aggregation method.

B.1 PATE-based Method

PATE (Papernot et al., 2016) was originally crafted for addressing classification tasks, and it incorporates both a private dataset and a public unlabeled dataset within its framework, drawing parallels to the principles of semi-supervised learning. PATE ensures DP by employing a teacher-student knowledge distillation framework consisting of multiple teacher models and a student model. In this setup, the student model acquires knowledge from the private dataset through knowledge distillation facilitated by the teacher models. The PATE framework consists of three key components: (i) Teacher Model Training: The private dataset is first shuffled and divided into M distinct subsets. Each teacher model is subsequently trained on one of these subsets. (ii) Teacher Aggregation: To leverage the knowledge of the individual teacher models, their outputs are aggregated, and this aggregated information serves as supervision for the student model. Each of the trained teachers contributes their insights to guide the learning process of the student on the unlabeled public dataset. (iii) Student Model Training: The student model is trained on the public dataset using the guidance provided by the aggregated teacher models. This collaborative approach ensures that the student model learns from the unlabeled data while benefiting from the distilled knowledge of the teacher models.

In the context of classification tasks, a common practice involves leveraging the collective wisdom of teachers by using their noisy majority votes as labels to guide the students, thereby ensuring DP. However, when it comes to text generation tasks, the straightforward application of this framework encounters a significant challenge. This challenge arises because traditional text generation models generate words sequentially, typically from left to right. Consequently, a straightforward application of PATE to text generation necessitates the iterative unveiling of all teachers, word by word, which comes with substantial computational and privacy costs. To tackle this issue, an innovative solution was presented by Tian et al. (2022), known as the SeqPATE framework. The SeqPATE framework initiates by generating pseudo-data using a pre-trained language model, simplifying the teachers' role to providing token-level guidance based on these pseudo inputs. In dealing with the inherent complexities of the expansive word output space and the accompanying noise, the framework introduces dynamic filtering of candidate words. This process focuses on selecting words with notably high probabilities. Additionally, the SeqPATE framework adopts a unique approach to aggregating teacher outputs. Instead of relying on voting, it involves an interpolation of their output distributions, offering a more refined and nuanced strategy for information fusion.

Recently, a notable development in the application of PATE, as reported by Duan et al. (2023), extends its utility to the realm of private In-context learning, a domain where the primary objective revolves around safeguarding the privacy of downstream data embedded in discrete prompts. Departing from the conventional approach of training teacher models on distinct partitions of private data, this innovative method capitalizes on the private data to formulate distinct prompts for the Large Language Model (LLM). In the context of private knowledge transfer, the teachers take on the role of labeling public data sequences. Each teacher offers their perspective by voting on the most probable class labels for the private downstream task. On the student model front, a novel strategy is proposed, leveraging the data efficiency of the prompting technique. This approach entails using labeled public sequences to create new discrete prompts for the student model. The chosen prompt is subsequently deployed alongside the Large Language Model (LLM) to serve as the student model, effectively enhancing the overall efficiency and privacy of the In-context learning process.

B.2 Sample-and-Aggregation-based Method

In contrast to the PATE-based method, the Sample-and-Aggregation-based approach diverges significantly by omitting the presence of a public unlabeled dataset, rendering the incorporation of a student model unnecessary. Notably, the work by Wu et al. (2023) delves into the realm of private In-context learning and provides a comprehensive protocol. The protocol encompasses the following crucial steps: The initial step involves the discreet partitioning of the dataset, specifically the private demonstration exemplars, into non-overlapping subsets of exemplars. Each of these subsets is then paired with relevant queries, culminating in the creation of exemplar-query pairs. For every exemplar-query pair, the Language Model's (LLM) API is invoked, eliciting a diverse set of responses. Subsequently, these individual responses generated by the LLM are aggregated in a manner compliant with differential privacy (DP) principles. The outcome is a privately aggregated model answer, which is then made available to the user. Furthermore, the study introduces two distinctive private aggregation schemes, thus enhancing the repertoire of options for preserving privacy in the context of In-context learning.

In a parallel exploration of private In-context learning, Tang et al. (2023) consider the scenarios involving an infinite number of queries. In lieu of generating private answers, their innovative approach revolves around the creation of synthetic few-shot demonstrations using the private dataset. This method involves augmenting each private subset with the information generated thus far, collectively contributing to the likelihood of generating the subsequent token. To mitigate the impact of noise prior to the private aggregation phase, the approach strategically curtails the vocabulary to include only tokens found within the top-K indices of the next-token probability. This is derived solely from the instructional content, entirely excluding any input from the private data. The probabilities associated with the next token generation, extracted from each individual subset, are then subjected to a private aggregation process, ensuring a nuanced and privacy-preserving amalgamation of information.

C Differential Privacy Preliminaries

Differential Privacy (DP) is a data post-processing technique, which guarantees data privacy by confusing the attacker. To be more specific, suppose there is one dataset noted as S, and we can get another dataset S' by changing or deleting one data record in this dataset. Denote the output distribution when S is the input as P_1, and the output distribution when S' is the input as P_2, if P_1 and P_2 are almost the same, then we cannot distinguish these two distributions, i.e., we cannot infer whether the deleted or replaced data sample based on the output we observed. The formal details are given by Dwork et al. (2006). Note that in the definition of DP, adjacency is a key notion. One of the commonly used adjacency definitions is that two datasets S and S' are adjacent (denoted as $S \sim S'$) if S' can be obtained by modifying one record in S.

Definition 2. Given a domain of dataset $\mathcal{X}$. A randomized algorithm $\mathcal{A} : \mathcal{X} \mapsto \mathcal{R}$ is (ε, δ)-differentially private (DP) if for all adjacent datasets S, S' with each sample is in $\mathcal{X}$ and for all $T \subseteq \mathcal{R}$, the following holds

$$\Pr(\mathcal{A}(S) \in T) \leq \exp(\varepsilon) \Pr(\mathcal{A}(S') \in T) + \delta.$$

When $\delta = 0$, we call the algorithm $\mathcal{A}$ is ε-DP.

Illustration: For example, let $\mathcal{X}$ be a collection of labeled product reviews, each belonging to a single individual, and let $\mathcal{R}$ be the parameters of a classifier trained on $\mathcal{X}$. If the classifier's training procedure $\mathcal{A}$ satisfies the DP definition above, an attacker's ability to find out whether a particular individual was present in the training data or not is limited by ε and δ.

In the definition of DP, there are two parameters ϵ and δ. Specifically, ϵ measures the closeness between the output distribution when the input is S, and the output distribution when the input is S', smaller ϵ indicates the two distributions are more indistinguishable, i.e., the algorithm $\mathcal{A}$ will be more private. In practice, we set $\epsilon = 0.1 - 0.5$ as a high privacy regime. Informally, δ could be thought of as the probability ratio between the two distributions is not bounded by e^ϵ. Thus, it is preferable to set δ as small as possible. In practice we always set δ as a value from $\frac{1}{n^{1.1}}$ to $\frac{1}{n^2}$, where n is the number of samples in the dataset S. It is notable that besides ϵ and (ϵ, δ)-DP, there are also other definitions DP such as Rényi DP (Mironov, 2017), Concentrated DP (Bun and Steinke, 2016; Dwork and Rothblum, 2016), Gaussian DP (Dong et al., 2022) and Truncated CDP (Bun et al., 2018). However, all of them can be transformed into the original definition of DP. Thus, in this survey, we mainly focus on Definition 2.

There are several important properties of DP, see (Dwork and Roth, 2014) for details. Here, we only introduce those which are commonly used in NLP tasks. The first one is post-processing, which means that any post-processing on the output of an (ϵ, δ)-DP algorithm will remain (ϵ, δ)-DP. Equivalently, if an algorithm is DP, then any side information available to the adversary cannot increase the risk of privacy leakage.

Proposition 1. Let $\mathcal{A} : \mathcal{X} \mapsto \mathbb{R}$ be (ϵ, δ)-DP, and let $f : \mathcal{R} \mapsto \mathcal{R}'$ be a (randomized) algorithm. Then $f \circ \mathcal{A} : \mathcal{X} \mapsto \mathbb{R}'$ is (ϵ, δ)-DP.

Example: Continuing with our scenario of training a review classifier under DP, let us imagine we take the model from the previous example, which was trained under (ε, δ)-DP, and perform a domain adaptation by fine-tuning on a different dataset, this time without any privacy. The resulting model still remains (ε, δ)-DP with respect to the original data, that is, privacy cannot be weakened by any post-processing.

The second property is the composition property. Generally speaking, the composition property guarantees that the composition of several DP mechanisms is still DP.

Proposition 2 (Basic Composition Theorem)**.** Let $\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_k$ be k sequence of randomized algorithms, where $\mathcal{A}_1 : \mathcal{X} \mapsto \mathcal{R}_1$ and $\mathcal{A}_i : \mathcal{R}_1 \times \cdots \mathcal{R}_{i-1} \times \mathcal{X} \mapsto \mathcal{R}_i$ for $i = 2, \cdots, k$. Suppose that for each $i \in [k]$, $\mathcal{A}_i(a_1, \cdots, a_{i-1}, \cdot)$ is (ϵ_i, δ_i)-DP. Then the algorithm $\mathcal{A} : \mathcal{X} \mapsto \mathcal{R}_1 \times \cdots \times \mathcal{R}_k$ that runs the algorithms $\mathcal{A}_i$ in sequence is (ϵ, δ)-DP with $\epsilon = \sum_{i=1}^{k} \epsilon_i$ and $\delta = \sum_{i=1}^{k} \delta_i$.

The basic composition allows us to design complex algorithms by putting together smaller pieces. We can view the overall privacy parameter ϵ as a budget to be divided among these pieces. We will thus often refer to (ϵ, δ) as the "privacy budget": each algorithm we run leaks some information, and consumes some of our budget. Differential privacy allows us to view information leakage as a resource to be managed. For example, if we fix the privacy budget (ϵ, δ), then making each $\mathcal{A}_i$ be $(\frac{\epsilon}{k}, \frac{\delta}{k})$-DP is sufficient to ensure the composition is (ϵ, δ)-DP.

Example: In most of the NLP tasks, we need to train a model by using variants of optimization methods, such as SGD or Adam. In general, these optimizers include several iterations to update the model, which could be thought of as a composition algorithm, and each iteration could be thought of as an algorithm. Thus, it is sufficient to design a DP algorithm for each iteration, and we can use the composition theorem to calculate the budget of the whole process.

Besides the basic composition property, there are also several advanced composition theorems for (ϵ, δ)-DP, which could provide tighter privacy guarantees than the basic one. For example, consider each $\mathcal{A}_i, i \in [k]$ is (ϵ, δ)-DP. Then the basic composition theorem implies their composition is $(k\epsilon, k\delta)$-DP. However, this is not tight as we can use the advanced composition theorem to show their composition could be improved to $(O(\sqrt{k}\epsilon, O(k\delta))$-DP (Dwork et al., 2010). We refer to reference (Kairouz et al., 2015; Murtagh and Vadhan, 2016; Meiser and Mohammadi, 2018) for details.

The third property is the privacy amplification via subsampling. Intuitively, every differentially private algorithm has a much lower privacy parameter ϵ when it is run on a secret sample than when it is run on a sample whose identities are known to the attacker. And there, a secret sample can be obtained by subsampling as it introduces additional randomness.

Proposition 3. Let A be an (ϵ, δ)-DP algorithm. Now we construct the algorithm B as follows: On input $D = \{x_1, \cdots, x_n\}$, first we construct a new sub-sampled dataset D_S where each $x_i \in D_s$ with probability q. Then we run algorithm A on the dataset D_S. Then $B(D) = A(D_S)$ is $(\tilde{\epsilon}, \tilde{\delta})$-DP, where $\tilde{\epsilon} = \ln(1 + (e^{\epsilon} - 1)q)$ and $\tilde{\delta} = q\delta$.

Example: The subsampling property can be used for the private version of the stochastic optimization method. As in these methods, a common strategy is to use the subsampled gradient to estimate the whole gradient.

It is notable that, besides subsampling, some other procedures could also amplify privacy, such as random check-in (Balle et al., 2020), mixing (Balle et al., 2019) and decentralization (Cyffers and Bellet, 2022). And for different subsampling methods, the privacy amplification guarantee is also different (Imola and Chaudhuri, 2021; Zhu and Wang, 2019; Balle et al., 2018).

In the following, we will introduce some mechanisms commonly used in NLP tasks to achieve DP.

We first give the definition of a (numeric) query. The query is simply something we want to learn from the dataset. Formally, a query could be any function f applied to a dataset S and outputting a real valued vector, formally $f : \mathcal{X} \mapsto \mathbb{R}^d$. For example, numeric queries might return the sum of the gradient of the loss on all samples, number of females in the database, or a textual summary of medical records of all persons in the database represented as a dense vector. Given a dataset S, a common paradigm for approximating $f(S)$ differentially privately is via adding some randomized noise. Laplacian noise and Gaussian noise are the most commonly used ones, which correspond to the Laplacian and Gaussian mechanisms, respectively.

Definition 3 (Laplacian Mechanism)**.** Given a query $f : \mathcal{X} \mapsto \mathbb{R}^d$, the Laplacian Mechanism is defined as: $\mathcal{M}_L(S, f, \epsilon) = q(S) + (Y_1, Y_2, \cdots, Y_d)$, where Y_i is i.i.d. drawn from a Laplacian Distribution $\text{Lap}(\frac{\Delta_1(f)}{\epsilon})$, where $\Delta_1(f)$ is the ℓ_1-sensitivity of the function f, *i.e.,* $\Delta_1(f) = \sup_{S' \sim S'} ||f(S) - f(S')||_1$. For a parameter λ, the Laplacian distribution has the density function $\text{Lap}(\lambda)(x) = \frac{1}{2\lambda}\exp(-\frac{x}{\lambda})$. Laplacian Mechanism preserves ϵ-DP.

Definition 4 (Gaussian Mechanism)**.** Given a query $f : \mathcal{X} \mapsto \mathbb{R}^d$, the Gaussian mechanism is defined as $\mathcal{M}_F(S, f, \epsilon, \delta) = q(S) + \xi$ where $\xi \sim \mathcal{N}(0, \frac{2\Delta_2^2(f)\log(1.25/\delta)}{\epsilon^2}\mathbb{I}_d)$, where $\Delta_2(f)$ is the ℓ_2-sensitivity of the function f, *i.e.,* $\Delta_2(f) = \sup_{S \sim S'} ||f(S) - f(S')||_2$. Gaussian mechanism preserves (ϵ, δ)-DP when $0 < \epsilon \leq 1$.

From the previous two mechanisms, we can see that to privately release $f(S)$, it is sufficient to calculate the ℓ_1-norm or ℓ_2-norm sensitivity first and add random noise. Moreover, as $\Delta_2(f) \leq \Delta_1(f)$, the Gaussian mechanism will have lower error than the Laplacian mechanism, while we relax the definition from ϵ-DP to (ϵ, δ)-DP.

Instead of answering $f(S)$ privately, we also always meet the selection problem, i.e., we want to output the best candidate among several candidates based on some score of the dataset. The exponential mechanism is the one that can output a nearly best candidate privately.

Definition 5 (Exponential Mechanism)**.** The Exponential Mechanism allows differentially private computation over arbitrary domains and range $\mathcal{R}$, parameterized by a score function $u(S, r)$ which maps a pair of input data set S and candidate result $r \in \mathcal{R}$ to a real-valued score. With the score function u and privacy budget ϵ, the mechanism yields an output with exponential bias in favor of high-scoring outputs. Let $\mathcal{M}(S, u, \mathcal{R})$ denote the exponential mechanism, and Δ be the sensitivity of u in the range $\mathcal{R}$, *i.e.,* $\Delta = \max_{r \in \mathcal{R}} \max_{D \sim D'} |u(D, r) - u(D', r)|$. Then if $\mathcal{M}(S, u, R)$ selects and outputs an element $r \in \mathcal{R}$ with probability proportional to $\exp(\frac{\epsilon u(S,r)}{2\Delta u})$, it preserves ϵ-DP.

In the original definition of DP, we assume that data are managed by a trusted centralized entity that is responsible for collecting them and for deciding which differentially private data analysis to perform and to release. A classical use case for this model is the one of census data. Compared with the above model (which is called the central model), there is another model, namely the local DP model, where each individual manages his/her proper data and discloses them to a server through some differentially private mechanisms. The server collects the (now private) data of each individual and combines them into a resulting data analysis. A classical use case for this model is the one aiming at collecting statistics from user devices like in the case of Google's Chrome browser. Formally, it is defined as follows.

Definition 6. For a data domain $\mathcal{X}$, a randomized algorithm $\mathcal{A} : \mathcal{X} \mapsto \mathcal{R}$ is called (ε, δ)-local DP (LDP) if for any $s, s' \in \mathcal{X}$ and $T \subseteq \mathcal{R}$ we have

$$\text{Pr}[\mathcal{A}(s) \in T] \leq e^{\varepsilon}\text{Pr}[\mathcal{A}(s') \in T] + \delta.$$

Compared with Definition 2, we can see that here the main difference is the inequality holds for all elements $s, s' \in \mathcal{X}$ instead of all adjacent pairs of the dataset. In this case, each individual could ensure that their own disclosures are DP via the randomizer $\mathcal{A}$. In some sense, the trust barrier is moved closer to the user. While this has the benefit of providing a stronger privacy guarantee, it also comes at a cost in terms of accuracy.

It is notable that besides the central DP and local DP model, there are also other intermediate models such as shuffle model (Cheu et al., 2019) and multiparty setting (Pathak et al., 2010). However, as they are seldom studied in NLP, we will not cover these protocols in this survey.