Differential Privacy Mechanisms in Neural Tangent Kernel Regression

Jiuxiang Gu[♡*] Yingyu Liang[♦,♥†] Zhizhou Sha[♣‡] Zhenmei Shi[♦§] Zhao Song[♠¶]
[♡]Adobe Research, USA. [♦]University of Wisconsin-Madison, USA.
[♥]The University of Hong Kong, HongKong. [♣]Tsinghua University, China.
[♠]The Simons Institute for the Theory of Computing at the University of California, Berkeley, USA.

Abstract

Training data privacy is a fundamental problem in modern Artificial Intelligence (AI) applications, such as face recognition, recommendation systems, language generation, and many others, as it may contain sensitive user information related to legal issues. To fundamentally understand how privacy mechanisms work in AI applications, we study differential privacy (DP) in the Neural Tangent Kernel (NTK) regression setting, where DP is one of the most powerful tools for measuring privacy under statistical learning, and NTK is one of the most popular analysis frameworks for studying the learning mechanisms of deep neural networks. In our work, we can show provable guarantees for both differential privacy and test accuracy of our NTK regression. Furthermore, we conduct experiments on the basic image classification dataset CIFAR10 to demonstrate that NTK regression can preserve good accuracy under a modest privacy budget, supporting the validity of our analysis. To our knowledge, this is the first work to provide a DP guarantee for NTK regression.

1. Introduction

Artificial Intelligence (AI) applications are widely employed in daily human life and product activities, such as face recognition [104], recommendation systems [135], chat-based language generation [2], and many more. These applications intrinsically run deep-learning models that are trained on broad datasets, where many contain sensitive user information, e.g., a company trains a model on its user information to provide better-customized service. Consequently, there is a problem with user privacy information data leakage [74], which affects the AI company's reputation [73] and may cause severe legal issues [66, 134]. Therefore, preserving the privacy of training data becomes a fundamental problem in deep learning.

Differential Privacy (DP) [36] was proposed to measure privacy rigorously and has been widely studied in many traditional statistical problems. Recently, many brilliant works have applied this powerful tool to machine learning settings. One line of work studies the classic machine learning task, e.g., [107] studies DP under the Support Vector Machine (SVM) model. However, these settings are still far from practical deep neural networks nowadays. The other line of work studies the DP in deep learning training, e.g., DP-SGD [1] provides DP guarantees for the Stochastic Gradient Descent (SGD) training algorithm. The issue in DP training is that the trade-off between privacy guarantees and test accuracy will worsen as training time increases. DP training may not be practical in today's training paradigm, i.e., pre-training with an adaptation in foundation models [16], as the per-training stage may involve billions of training steps.

To bridge the gap between practical deep learning models and practical differential privacy guarantees, in this work, we study DP Mechanisms in Neural Tangent Kernel [64] (NTK) Regression. NTK is one of the most standard analysis frameworks for studying optimization and generalization in over-parameterized deep neural networks (DNN). NTK can connect the DNN training by SGD to kernel methods by using the kernel induced by gradient around the neural networks' initialization. Consequently, the DNN optimization can be viewed as NTK regression, and it retains almost the same generalization ability [11] as kernel regression even though the DNN is in an over-parameterization regime.

Our contributions. In this work, we use the "Gaussian Sampling Mechanism" [46] to add a positive semi-definite noise matrix to the neural tangent kernel matrix and the truncated Laplace mechanism [36] to ensure the privacy of the kernel function. Then, we can show provable guarantees for both differential privacy and test accuracy of our private NTK regression, i.e.,

Theorem 1.1 (Main result, informal version of Theorem 4.4)**.** *Under proper conditions, for any test data x, we*

*jigu@adobe.com.
†yingyul@hku.hk. yliang@cs.wisc.edu.
‡shazz20@mails.tsinghua.edu.cn.
§zhmeishi@cs.wisc.edu.
¶magic.linuxkde@gmail.com.

have NTK-regression is (ϵ, δ)-DP and has good utility under a large probability.

Furthermore, we undertake experiments using the fundamental image classification dataset CIFAR10 to illustrate that NTK regression can maintain high accuracy with a modest privacy budget (see Figure 1 in Section 6.2). This effectively validates our analysis. To the best of our knowledge, this research is the first effort to offer a differential privacy (DP) guarantee for NTK regression.

Roadmap. Our paper is organized as follows. Section 2 provides an overview of differential privacy, the neural tangent kernel (NTK). Section 3 introduces the formal definition of DP, the definitions of both continuous and discrete versions of the NTK matrix, and the definition of NTK Regression. In Section 4, we provide the privacy and utility guarantees for our private NTK regression. An overview of the techniques employed in this paper is discussed in Section 5. In Section 6, we conduct experiments on the ten-class classification task of the CIFAR-10 dataset, demonstrating that our algorithm preserves good utility and privacy. In Section 7, we thoroughly discuss several inherent intuitions behind the design of our algorithm. Finally, we conclude in Section 8.

2. Related Work

Differential Privacy Guarantee Analysis. Since the introduction of the concept of differential privacy (DP) by [35], it has emerged as a crucial standard for privacy protection, both theoretically and empirically [34,75,105,132,136]. DP offers a robust and measurable privacy definition that supports the design of algorithms with specific guarantees of privacy and accuracy [8,13,22,24,27–31,38–43,46,49,51–53, 60, 63, 67, 76, 77, 79–81, 88, 98, 99, 114, 117, 120, 129] and many more. Furthermore, innovative mechanisms have been developed beyond conventional Laplace, Gaussian, and Exponential methods [36]. For instance, the truncated Laplace mechanism [48] has been demonstrated to achieve the tightest lower and upper bounds on minimum noise amplitude and power among all (ϵ, δ)-DP distributions.

Neural Tangent Kernel. Numerous recent studies suggest that the analysis of optimization and generalization in deep learning should be closely integrated. NTK employs the first-order Taylor expansion to examine highly over-parameterized neural networks, from their initial states, as seen in references like [4–6, 10, 17–20, 23, 25, 26, 32, 33, 44, 45, 47, 64, 65, 70, 71, 83, 90, 93, 94, 96, 97, 100–103, 106, 111, 112, 116, 118, 119, 131, 138–141] and others. Consequently, the optimization of neural networks can be approached as a convex problem. The NTK technique has gained widespread application in various contexts, including preprocessing analysis [7,45,62,109,115,121,122], federated learning [78], LoRA adaptation for large language models [21, 55, 56, 61, 84, 85, 91, 110, 130], and estimating scoring functions in diffusion models [54, 57, 58, 123–126].

Differential Privacy in Machine Learning. Differential privacy (DP) is a thriving and potent method with extensive applications in the field of private machine learning [87]. This includes its use during the pretraining stage [1, 105], the adaptation stage [14,82,89,108,113,133], and the inference stage [37,72,86]. Recently, [50,92,137] has integrated DP into large language models, and [128] applied DP into diffusion models. DP has also been widely used in various settings, e.g., near-neighbor counting [8], permutation hashing [80], BDD tree [63], counting tree [49], Jaccard similarity [12] and so on.

3. Preliminary

In this section, we first introduce some basic notations in Section 3.1. In Section 3.2, we introduce the definition of DP, and the truncated Laplace mechanism. Then, we will introduce the definitions of the neural tangent kernel, both discrete and continuous versions in Section 3.3, the definitions and key component of NTK regression in Section 3.4.

3.1. Notations

For any positive n, let $[n]$ denote the set $\{1, 2, \cdots, n\}$. For any vector $z \in \mathbb{R}^n$. We define the ℓ_2-norm of a vector z as $\|z\|_2 := (\sum_{i=1}^n z_i^2)^{1/2}$, the ℓ_1-norm as $\|z\|_1 := \sum_{i=1}^n |z_i|$, the ℓ_0-norm as the count of non-zero elements in z, and the ℓ_∞-norm as $\|z\|_\infty := \max_{i \in [n]} |z_i|$. The transpose of vector z is indicated by $z^\top$. The inner product between two vectors is denoted by $\langle \cdot, \cdot \rangle$, such that $\langle a, b \rangle = \sum_{i=1}^n a_i b_i$.

For any matrix $A \in \mathbb{R}^{m \times n}$. We define the Frobenius norm of A as $\|A\|_F := (\sum_{i \in [m], j \in [n]} A_{i,j}^2)^{1/2}$. We use $\|A\|$ to denote the spectral/operator norm of matrix A.

A function $f(x)$ is said to be L-Lipschitz continuous if it satisfies the condition $\|f(x) - f(y)\|_2 \leq L \cdot \|x - y\|_2$ for some constant L. Let $\mathcal{D}$ represent a given distribution. The notation $x \sim \mathcal{D}$ indicates that x is a random variable drawn from the distribution $\mathcal{D}$. We employ $\mathbb{E}[]$ to represent the expectation operator and $\Pr[]$ to denote probability. Furthermore, we refer to a matrix as PSD to indicate that it is positive semi-definite.

As we have multiple indexes, to avoid confusion, we usually use $i, j \in [n]$ to index the training data, $s, t \in [d]$ to index the feature dimension, $r \in [m]$ to index neuron number.

3.2. Differential Privacy

This section will first introduce the formal definition of differential privacy. Then, we will introduce the truncated Laplace mechanism that can ensure DP.

Definition 3.1 (Differential Privacy, [36]). *For $\epsilon > 0, \delta \geq 0$, a randomized function $\mathcal{A}$ is (ϵ, δ)-differentially private ((ϵ, δ)-DP) if for any two neighboring datasets $X \sim X'$, and any possible outcome of the algorithm $S \subset$ Range($\mathcal{A}$), $\Pr[\mathcal{A}(X) \in S] \leq e^{\epsilon} \Pr[\mathcal{A}(X') \in S] + \delta$.*

Then, we introduce the sensitivity of a function f, which is defined to be $\Delta_f = \max_{X \sim X'} |f(X) - f(X')|$. We use $X \sim X'$ to denote two neighboring datasets.

We use $\mathrm{Lap}(\lambda)$ to denote the Laplace distribution with parameter λ with PDF $\Pr[Z = z] = \frac{1}{2\lambda} e^{-|z|/\lambda}$. We also use $\mathrm{TLap}(\Delta, \epsilon, \delta)$ to denote the Truncated Laplace distribution with PDF proportional to $e^{-|z|/\lambda}$ on the region $[-B_L, B_L]$, where $B_L = (\Delta/\epsilon) \log(1 + \frac{e^{\epsilon}-1}{2\delta})$.

Lemma 3.2 (Truncated Laplace Mechanism, [8, 36, 48]). *Given a numeric function f that takes a dataset X as the input, and has sensitivity Δ, the mechanism output $f(X) + Z$ where $Z \sim \mathrm{Lap}(\Delta/\epsilon)$ is $(\epsilon, 0)$-DP. In addition, if $Z \sim \mathrm{TLap}(\Delta, \epsilon, \delta)$, then $f(X) + Z$ is (ϵ, δ)-DP.*

Here, we introduce the critical post-processing Lemma for DP.

Lemma 3.3 (Post-Processing Lemma for DP, [36]). *Let $\mathcal{M} := \mathbb{N}^{|\chi|} \to \mathbb{R}$ be a randomized algorithm that is (ϵ, δ)-differentially private. Let $f : \mathbb{R} \to \mathbb{R}'$ be an arbitrarily random mapping. Then is $f \circ \mathcal{M} : \mathbb{N}^{|\chi|} \to \mathbb{R}'$ (ϵ, δ)-differentially private.*

Then, we restate the composition lemma for DP.

Lemma 3.4 (Composition lemma for DP, [36]). *Let $\mathcal{M} := \mathbb{N}^{|\chi|} \to \mathbb{R}$ be an (ϵ_i, δ_i)-DP algorithm for $i \in [k]$. If $\mathcal{M}_{[k]} \to \Pi_{i=1}^n \mathcal{R}_i$ satisfies $\mathcal{M}_{[k]}(x) = (\mathcal{M}_1(x), \cdots, \mathcal{M}_k(x))$, then $\mathcal{M}_{[k]}$ is $(\sum_{i=1}^k \epsilon_i, \sum_{i=1}^k \delta_i)$-DP.*

3.3. Neural Tangent Kernel

Then, we introduce our crucial concept, the Neural Tangent Kernel induced by the Quadratic activation function. We will introduce Discrete Quadratic Kernel in Definition 3.5 and Continuous Quadratic Kernel in Definition 3.6.

Data. We have n training data points $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n = (X, Y)$, where $x \in \mathbb{R}^d$ and $y \in \{-1, 1\}$. We denote $X = [x_1, \ldots, x_n]^\top \in \mathbb{R}^{n \times d}$ and $Y = [y_1, \ldots, y_n]^\top \in \{-1, +1\}^n$. We assume that $\|x_i\|_2 \leq B$, $\forall i \in [n]$.

Models. We consider the two-layer neural network with quadratic activation function and m neurons

$$f(x) = \sum_{r=1}^m a_r \langle w_r, x \rangle^2,$$

where $w_r \in \mathbb{R}^d$ and $a_r \in \{-1, +1\}$ for any $r \in [m]$.

Definition 3.5 (Discrete Quadratic NTK Kernel). *We draw weights $w_r \sim \mathcal{N}(0, \sigma^2 I_{d \times d})$ for any $r \in [m]$ and let them be fixed. Then, we define the discrete quadratic kernel matrix $H^{\mathrm{dis}} \in \mathbb{R}^{n \times n}$ corresponding to $\mathcal{D}_n$, such that $\forall i, j \in [n]$, we have*

$$H_{i,j}^{\mathrm{dis}} = \frac{1}{m} \sum_{r=1}^m \langle \langle w_r, x_i \rangle x_i, \langle w_r, x_j \rangle x_j \rangle.$$

Note that H^{dis} is a PSD matrix, where a detailed proof can be found in Lemma C.1.

Definition 3.6 (Continuous Quadratic NTK Kernel). *We define the continuous quadratic kernel matrix $H^{\mathrm{cts}} \in \mathbb{R}^{n \times n}$ corresponding to $\mathcal{D}$, such that $\forall i, j \in [n]$, we have*

$$H_{i,j}^{\mathrm{cts}} = \mathop{\mathbb{E}}_{w \sim \mathcal{N}(0, \sigma^2 I_{d \times d})} \langle \langle w, x_i \rangle x_i, \langle w, x_j \rangle x_j \rangle.$$

3.4. NTK Regression

We begin by defining the classical kernel regression problem as follows:

Definition 3.7 (Classical kernel ridge regression [71]). *Let feature map $\phi : \mathbb{R}^d \to \mathcal{F}$ and $\lambda > 0$ is the regularization parameter. A classical kernel ridge regression problem can be written as*

$$\min_{w \in \mathbb{R}^n} \frac{1}{2} \|Y - \phi(X)^\top w\|_2^2 + \frac{1}{2} \lambda \|w\|_2^2.$$

Then, we are ready to introduce the NTK Regression problem as follows:

Definition 3.8 (NTK Regression [71]). *If the following conditions are met:*

- *Let $\mathsf{K}(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a kernel function, i.e., $\mathsf{K}(x, z) = \frac{1}{m} \sum_{r=1}^m \langle \langle w_r, x \rangle x, \langle w_r, z \rangle z \rangle, \forall x, z \in \mathbb{R}^d$.*
- *Let $K \in \mathbb{R}^{n \times n}$ be the kernel matrix with $K_{i,j} = \mathsf{K}(x_i, x_j)$, $\forall i, j \in [n] \times [n]$.*
- *Let $\alpha \in \mathbb{R}^n$ be the solution to $(K + \lambda I_n)\alpha = Y$. Namely, we have $\alpha = (K + \lambda I_n)^{-1} Y$.*

Then, for any data $x \in \mathbb{R}^d$, the NTK Kernel Regression can be denoted as

$$f_K^*(x) = \frac{1}{n} \mathsf{K}(x, X)^\top \alpha.$$

4. Main Results

In this section, we will introduce several essential lemmas that form the basis of our main result.

Firstly, we provide a high-level overview of our intuition. Recall that in the definition of the NTK regression, we have $\alpha = (K + \lambda I)^{-1} Y$ and $\mathsf{K}(x, X)$ (see also Definition 3.8).

We aim to protect the sensitive information in the training data $X \in \mathbb{R}^{n \times d}$. Therefore, we only need to ensure the privacy of α and $\mathsf{K}(x, X)$.

To ensure the privacy of α, we initially focus on privatizing K. Subsequently, we demonstrate that α remains private by applying the post-processing lemma of DP (Lemma 3.3). Privatizing K is non-trivial, as $K = H^{\mathrm{dis}}$, indicating that K is a positive semi-definite (PSD) matrix (Lemma C.1). We denote $\widetilde{K}$ as the privacy-preserving counterpart of K. $\widetilde{K}$ must maintain the PSD property, a condition that classical mechanisms such as the Laplace Mechanism and Gaussian Mechanism cannot inherently guarantee a private matrix. In the work by [46], the "Gaussian Sampling Mechanism" addresses this challenge, ensuring that the private version of the Attention Matrix also retains PSD.

Then, we use the truncated Laplace mechanism to ensure the privacy of X. Then, by post-processing lemma, we have the privacy guarantees for $\mathsf{K}(x, X)$.

Finally, with the help of the composition lemma of DP, we can show that the entire NTK regression is also DP.

So far, we have introduced the high-level intuition of our entire algorithm. Then, we will dive into the details. Firstly, we introduce the definition of the neighboring datasets.

Definition 4.1 (β-close neighbor dataset, [46]). *Let $B > 0$ be a constant. Let n be the number of data points. Let dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$ and $\|x_i\|_2 \leq B$ for any $i \in [n]$. We define $\mathcal{D}'$ as a neighbor dataset with one data point replacement of $\mathcal{D}$. Without loss of generality, we have $\mathcal{D}' = \{(x_i, y_i)\}_{i=1}^{n-1} \cup \{(x_n', y_n)\}$.Namely, we have $\mathcal{D}$ and $\mathcal{D}'$ only differ in the n-th item.*

Additionally, we assume that x_n and x_n' are β-close. Namely, we have

$$\|x_n - x_n'\|_2 \leq \beta.$$

4.1. DP Guarantees for NTK Regression

In this section, we will state the DP property of the entire NTK regression using the composition and post-processing lemma of DP. The corresponding lemma is as follows.

Lemma 4.2 (DP guarantees for NTK regression , informal version of Lemma I.1). *Let $\mathsf{K}(x, \widetilde{X})$ be defined as Lemma 5.9. Let $(\widetilde{K} + \lambda I)^{-1}$ be defined as Lemma 5.4. Let $\epsilon_X, \delta_X \in \mathbb{R}$ denote the DP parameter for $\mathsf{K}(x, X)$. Let $\epsilon_\alpha, \delta_\alpha \in \mathbb{R}$ denote the DP parameter for $(K + \lambda I)^{-1}$. Let $\epsilon = \epsilon_X + \epsilon_\alpha, \delta = \delta_X + \delta_\alpha$. Then, we can show that the private NTK regression (Algorithm 2) is (ϵ, δ)-DP.*

Basically, we can easily prove this lemma by using the composition lemma of DP. For further details of the proof, please refer to Section I in the appendix.

4.2. Utility Guarantees for NTK Regression

In this section, we provide the utility guarantees for the private NTK regression introduced in the previous section.

Lemma 4.3 (Utility guarantees for NTK regression, informal version of Lemma J.1). *Let $\Delta_X = \sqrt{d} \cdot \beta$. Let $\epsilon_X, \delta_X \in \mathbb{R}$ denote the DP parameters for X. Let $B_L = (\Delta_X / \epsilon_X) \log(1 + \frac{e^{\epsilon_X} - 1}{2\delta_X})$. If all conditions hold in Condition 5.6, then, with probability $1 - \gamma$, we have*

$$|f_K^*(x) - f_{\widetilde{K}}^*(x)| \leq O(\frac{B^3 \sqrt{d} B_L}{\eta_{\min} + \lambda} + \frac{\rho \cdot \eta_{\max} \cdot \omega}{(\eta_{\min} + \lambda)^2}).$$

4.3. Main Theorem

Then, we are ready to introduce our main result, including both the privacy and utility guarantees of our private NTK regression (Algorithm 2).

Theorem 4.4 (Private NTK regression). *Let $\Delta_X = \sqrt{d} \cdot \beta$. Let $\epsilon_X, \delta_X \in \mathbb{R}$ denote the DP parameters for X. Let $B_L = (\Delta_X / \epsilon_X) \log(1 + \frac{e^{\epsilon_X} - 1}{2\delta_X})$. If all conditions hold in Condition 5.3, Condition 5.5, and Condition 5.6, then, for any test data x, with probability $1 - \delta_3 - \gamma$, we have that $f_{\widetilde{K}}^*(x)$ is (ϵ, δ)-DP and*

$$|f_K^*(x) - f_{\widetilde{K}}^*(x)| \leq O(\frac{B^3 \sqrt{d} B_L}{\eta_{\min} + \lambda} + \frac{\rho \cdot \eta_{\max} \cdot \omega}{(\eta_{\min} + \lambda)^2}).$$

The proof of this theorem follows from directly combining the DP guarantees of NTK regression (Lemma 4.2) and the utility guarantees of NTK regression (Lemma 4.3).

5. Technical Overview

In Section 5.1, we introduce two crucial concepts used for the Gaussian sampling mechanism. In Section 5.2, we will adhere to the framework established in [46] and elaborate on the functioning of the "Gaussian Sampling Mechanism," along with its primary results and requirements. In Section 5.3, we will examine the utility implications of employing the "Gaussian Sampling Mechanism." Additionally, we include a remark that analyzes the trade-off between privacy and utility inherent in our approach. In Section 5.4, we introduce our privacy guarantee results on the kernel function $\mathsf{K}(x, X)$, which is achieved by the truncated Laplace mechanism. In Section 5.5, we provide a detailed analysis of the utility of the private kernel function $\mathsf{K}(x, \widetilde{X})$.

5.1. Key Concepts

In this section, we will introduce the two essential definitions M and Δ used in the privacy proof of "Gaussian Sampling Mechanism", which need to satisfy $M < \Delta$, which is also the **Condition 4** in Condition 5.3. We will begin by presenting the definition of M.

Definition 5.1 (Definition of M, [46]). *Let $\mathcal{M} : (\mathbb{R}^n)^d \to \mathbb{R}^{n \times n}$ be a (randomized) algorithm that given a dataset of d points in $\mathbb{R}^n$ outputs a PSD matrix. Let $\mathcal{Y}, \mathcal{Y}' \in (\mathbb{R}^n)^d$. Then, we define*

$$M := \|\mathcal{M}(\mathcal{Y})^{1/2} \mathcal{M}(\mathcal{Y}')^{-1} \mathcal{M}(\mathcal{Y})^{1/2} - I\|_F.$$

Afterward, we proceed to define Δ.

Definition 5.2 (Definition of Δ, [46]). *If we have the following conditions:*

- *Let $\epsilon \in (0,1)$ and $\delta \in (0,1)$.*
- *Let k denote the number of i.i.d. samples $g_1, g_2, \cdots, g_k$ from $\mathcal{N}(0, \Sigma_1)$ output by Algorithm 1.*

We define

$$\Delta := \min \left\{ \frac{\epsilon}{\sqrt{8k \log(1/\delta)}}, \frac{\epsilon}{8 \log(1/\delta)} \right\}.$$

5.2. DP Guarantees for $(K + \lambda I)^{-1}$

In this section, we recapitulate the analytical outcomes of the "Gaussian Sampling Mechanism" as presented in Theorem 5.4 from [46]. The associated algorithm is detailed in Algorithm 1.

Firstly, we outline the conditions employed in the "Gaussian Sampling Mechanism" as follows:

Condition 5.3. *We need the following conditions for DP:*

- **Condition 1.** *Let $\epsilon_\alpha \in (0,1)$, $\delta_\alpha \in (0,1)$, $k \in \mathbb{N}$.*
- **Condition 2.** *Let $\mathcal{Y}, \mathcal{Y}'$ denote neighboring datasets, which differ by a single data element.*
- **Condition 3.** *Let Δ be defined in Definition 5.2 and $\Delta < 1$.*
- **Condition 4.** *Let $M, \mathcal{M}$ be defined in Definition 5.1 and $M \leq \Delta$.*
- **Condition 5.** *Let the input $\Sigma = \mathcal{M}(\mathcal{Y})$.*
- **Condition 6.** *Let $\rho = O(\sqrt{(n^2 + \log(1/\gamma))/k} + (n^2 + \log(1/\gamma))/k)$.*

Prior to delving into the primary analysis of the "Gaussian Sampling Mechanism" , we offer a succinct overview of its underlying intuition. As noted at the outset of Section 4, the task of obtaining a private positive semi-definite (PSD) matrix is non-trivial.

Nevertheless, by leveraging covariance estimation within the "Gaussian Sampling Mechanism", we can guarantee that the estimated matrix will remain PSD. This is because for any $i \in [k]$, we have $g_i g_i^\top$ is PSD matrix, then $k^{-1} \sum_{i=1}^k g_i g_i^\top$ is also PSD matrix.

With this foundation, we are ready to introduce the analysis of the "Gaussian Sampling Mechanism". The analysis is presented as follows:

Lemma 5.4 (DP guarantees for $(K + \lambda I)^{-1}$, Theorem 6.12 in [46], Theorem 5.1 in [3], informal version of Lemma D.1). *If all conditions hold in Condition 5.3 and Condition 5.5, then, there exists an Algorithm 1 such that*

- *Part 1. Algorithm 1 is $(\epsilon_\alpha, \delta_\alpha)$-DP.*
- *Part 2. Outputs $\widehat{\Sigma} \in \mathbb{S}_+^n$ denotes the private version of input Σ, such that with probabilities at least $1 - \gamma$,*

$$\|\Sigma^{-1/2} \widehat{\Sigma} \Sigma^{-1/2} - I_n\|_F \leq \rho.$$

- *Part 3. $(1 - \rho)\Sigma \preceq \widehat{\Sigma} \preceq (1 + \rho)\Sigma$.*

In Lemma 5.4, **Part 1** claims the privacy guarantees of the "Gaussian Sampling Mechanism", **Part 2** establishes the critical properties necessary to ensure the utility of the "Gaussian Sampling Mechanism", and **Part 3** presents the ultimate utility outcomes of the algorithm.

Note that in our setting, we use $\Sigma = K$, where K is non-private Discrete Quadratic NTK Matrix in NTK Regression, and we also have $\widehat{\Sigma} = \widetilde{K}$, where $\widetilde{K}$ denotes the private version of K.

Algorithm 1 The Gaussian Sampling Mechanism, [46]

procedure ALGORITHM(Σ, k)
  PSD matrix $\Sigma \in \mathbb{R}^{n \times n}$ and parameter $k \in \mathbb{N}$
  Obtain vectors $g_1, g_2, \cdots, g_k$ by sampling $g_i \sim \mathcal{N}(0, \Sigma)$, independently for each $i \in [k]$
  Compute $\widehat{\Sigma} = \frac{1}{k} \sum_{i=1}^k g_i g_i^\top$ ▷ Covariance estimate.
  return $\widehat{\Sigma}$
end procedure

We need the following conditions so that we can make **Condition 4** in Condition 5.3 hold, with probability $1 - \delta_3$. See the detailed proof in Section B.3.

Condition 5.5. *We need the following conditions for the calculation of M (see Definition 5.1).*

- **Condition 1.** *If $\mathcal{D} \in \mathbb{R}^{n \times d}$ and $\mathcal{D}' \in \mathbb{R}^{n \times d}$ are neighboring dataset (see Definition 4.1)*
- **Condition 2.** *Let H^{dis} denote the discrete NTK kernel matrix generated by $\mathcal{D}$, and $H^{\mathrm{dis}'}$ denotes the discrete NTK kernel matrix generated by neighboring dataset $\mathcal{D}'$.*
- **Condition 3.** *Let $H^{\mathrm{dis}} \succeq \eta_{\min} I_{n \times n}$, for some $\eta_{\min} \in \mathbb{R}$.*
- **Condition 4.** *Let $\beta = O(\eta_{\min} / \operatorname{poly}(n, \sigma, B))$, where β is defined in Definition 4.1.*
- **Condition 5.** *Let $\psi := O(\sqrt{n} \sigma^2 B^3 \beta)$.*

- **Condition 6.** *Let* $\delta_1, \delta_2, \delta_3 \in (0,1)$. *Let* $\delta_1 = \delta_2 / \operatorname{poly}(m)$. *Let* $\delta_2 = \delta_3 / \operatorname{poly}(n)$.
- **Condition 7.** *Let* $d = \Omega(\log(1/\delta_1))$.
- **Condition 8.** *Let* $m = \Omega(n \cdot dB^2 \beta^{-2} \log(1/\delta_2))$.

5.3. Utility Guarantees for $(K + \lambda I)^{-1}$

In this section, we will provide utility guarantees under "Gaussian Sampling Mechanism". By Lemma 5.7, we will argue that, "Gaussian Sampling Mechanism" provides good utility under differential privacy.

We start with introducing the necessary conditions used in proving the utility of "Gaussian Sampling Mechanism".

Condition 5.6. *We need the following conditions for Utility guarantees of "Gaussian Sampling Mechanism":*

- **Condition 1.** *If* $\mathcal{D} \in \mathbb{R}^{n \times d}$ *and* $\mathcal{D}' \in \mathbb{R}^{n \times d}$ *are neighboring dataset (see Definition 4.1)*
- **Condition 2.** *Let* H^{dis} *denote the discrete NTK kernel matrix generated by* $\mathcal{D}$ *(see Definition 3.5).*
- **Condition 3.** *Let* $\eta_{\max} I_{n \times n} \succeq H^{\mathrm{dis}} \succeq \eta_{\min} I_{n \times n}$, *for some* $\eta_{\max}, \eta_{\min} \in \mathbb{R}$.
- **Condition 4.** *Let* $\widetilde{H}^{\mathrm{dis}}$ *denote the private* H^{dis} *generated by Algorithm 1 with* H^{dis} *as the input.*
- **Condition 5.** *Let* $K = H^{\mathrm{dis}}, \widetilde{K} = \widetilde{H}^{\mathrm{dis}}$ *in Definition 3.8. Then we have* $f^*_K(x)$ *and* $f^*_{\widetilde{K}}(x)$.
- **Condition 6.** *Let* $\sqrt{n}\psi/\eta_{\min} < \Delta$, *where* Δ *is defined in Definition 5.2.*
- **Condition 7.** *Let* $\rho = O(\sqrt{(n^2 + \log(1/\gamma))/k} + (n^2 + \log(1/\gamma))/k)$.
- **Condition 8.** *Let* $\omega := 6d\sigma^2 B^4$.
- **Condition 9.** *Let* $\gamma \in (0,1)$.

We then leverage **Part 3** of Lemma 5.4 to derive the error between the outputs of the private and non-private NTK Regression, thereby demonstrating the utility of our algorithm.

Lemma 5.7 (Utility guarantees for $(K + \lambda I)^{-1}$, informal version of Lemma F.3)**.** *If all conditions hold in Condition 5.6, then, with probability* $1 - \gamma$, *we have*

$$\|(K + \lambda I)^{-1} - (\widetilde{K} + \lambda I)^{-1}\| \leq O(\frac{\rho \cdot \eta_{\max}}{(\eta_{\min} + \lambda)^2})$$

The interplay between privacy and utility guarantees is complex. Our algorithm exhibits a property akin to that of other classical differential privacy algorithms: an increase in privacy typically results in a decrease in utility, and conversely. We will provide a thorough explanation of the privacy-utility trade-off in the subsequent Remark.

Algorithm 2 Private NTK Regression

procedure MAIN($X \in \mathbb{R}^{n \times d}, m, k$) ▷ Theorem 4.4
  Draw $w_1, \cdots, w_m \in \mathbb{R}^d$ random Gaussian vectors
  Compute K such such $K_{i,j} = \mathsf{K}(x_i, x_j)$
  Obtain vectors $g_1, g_2, \cdots, g_k$ by sampling $g_i \sim \mathcal{N}(0, K)$ independently for each $i \in [k]$
  Compute $\widetilde{K} \leftarrow \frac{1}{k} \sum_{i=1}^k g_i g_i^\top$ ▷ Lemma 5.4
  Compute $\widetilde{X} \leftarrow X + \mathrm{TLap}(\Delta_X, \epsilon_X, \delta_X)$ ▷ Lemma 5.9
  Compute $f^*_{\widetilde{K}}(x) \leftarrow \mathsf{K}(x, \widetilde{X})^\top (\widetilde{K} + \lambda \cdot I_n)^{-1} Y$
  return $f^*_{\widetilde{K}}(x)$
end procedure

Remark 5.8 (Trade-off between Privacy and Utility in Lemma 5.7)**.** *An inherent trade-off exists between the privacy and utility guarantees of our algorithm. Specifically, enhancing privacy typically results in a degradation of utility. Recall that the variable* k *represents the number of sampling iterations in the Gaussian Sampling Mechanism.*

To ensure privacy, we must adhere to **Condition 4** *as outlined in Condition 5.3, which requires that* $M < \Delta$. *Here,* M *is a constant defined in Definition 5.1, with its precise value calculated in Lemma B.1. In contrast,* Δ *is defined in Definition 5.2 and is dependent on the value of* k. *Consequently, to achieve stronger privacy, namely a smaller DP parameter* ϵ_α *or* δ_α, *it is necessary to decrease* k *to meet the* $M < \Delta$ *constraint.*

On the other hand, for utility considerations, as defined by $\rho = O(\sqrt{(n^2 + \log(1/\gamma))/k} + (n^2 + \log(1/\gamma))/k)$, *a reduction in* k *results in an increase in* ρ. *This, in turn, leads to diminished utility.*

Due to the limitation of space, we refer the readers to Lemma F.3 in the appendix for the details of proof of Lemma 5.7. A detailed explanation of the trade-off between privacy and utility can be found in Remark 5.8.

5.4. DP Guarantees for $\mathsf{K}(x, X)$

Then, we will introduce how to ensure the DP property of the kernel function $\mathsf{K}(x, X)$ by using the truncated Laplace mechanism.

Lemma 5.9 (DP guarantees for $\mathsf{K}(x, X)$, informal version of Lemma G.3)**.** *If the following conditions hold:*

- *Let* $x \in \mathbb{R}^d$ *denote an arbitrary query.*
- *Let* $\epsilon_X, \delta_X \in \mathbb{R}$ *denote the DP parameters.*
- *Let* $\Delta_X := \sqrt{d} \cdot \beta$ *denote the sensitivity of* X.
- *Let* $\mathsf{K}(x, X)$ *be defined as Definition 3.8.*
- *Let* $\widetilde{X} := X + \mathrm{TLap}(\Delta_X, \epsilon_X, \delta_X)$ *denote the private version of* X, *where* $\widetilde{X}$ *is* (ϵ_X, δ_X)*-DP.*

Then, we can show that $\mathsf{K}(x, \widetilde{X})$ *is* (ϵ_X, δ_X)*-DP.*

To sum up, we first use the truncated Laplace mechanism to ensure the (ϵ_X, δ_X)-DP on $\widetilde{X}$. Then, we use the post-processing lemma to ensure the privacy of $\mathsf{K}(x, X)$. More details can be found in Section G.

5.5. Utility Guarantees for $\mathsf{K}(x, X)$

The utility analysis for the private kernel function $\mathsf{K}(x, \widetilde{X})$ is as follows.

Lemma 5.10 (Utility guarantees for $\mathsf{K}(x, X)$, informal version of Lemma H.1)**.** *If the following conditions hold:*

- *Let* $x \in \mathbb{R}^d$ *be a query, where for some* $B \in \mathbb{R}$, $\|x\|_2 \leq B$.
- *Let* $\mathsf{K}(x, X) \in \mathbb{R}^n$ *be defined as Definition 3.8.*
- *Let* $\widetilde{X} \in \mathbb{R}^{n \times d}$ *be defined as Lemma G.2.*
- *Let* $\Delta_X = \sqrt{d} \cdot \beta$.
- *Let* $\epsilon_X, \delta_X \in \mathbb{R}$ *denote the DP parameters for* X.
- *Let* $B_L = (\Delta_X / \epsilon_X) \log(1 + \frac{\exp(\epsilon_X) - 1}{2\delta_X})$.

Then, we can show that

$$\|\mathsf{K}(x, \widetilde{X}) - \mathsf{K}(x, X)\|_2 \leq 2\sqrt{n} B^3 \sqrt{d} B_L.$$

6. Experiments

This section will introduce the experimental methodology employed on the CIFAR-10 dataset. The corresponding results are visualized in Fig. 1. In Section 6.1, we enumerate all the parameters and the experimental setup we utilized. Section 6.2 presents a detailed analysis of the outcomes from our experiments.

6.1. Experiment Setup

Dataset. Our experiments are conducted on the CIFAR-10 dataset [68], which comprises ten distinct classes of colored images, including subjects such as airplanes, cats, and dogs. The dataset is partitioned into 50,000 training samples and 10,000 testing samples, with each image measuring 32×32 pixels and featuring RGB channels. Although NTK regression is initially a binary classification model, we can extend it to ten classification classes. To be more specific, let n_{cls} denote the number of classes. Here, we have $n_{\mathrm{cls}} = 10$. Then, we have $Y \in \mathbb{R}^{n \times n_{\mathrm{cls}}}$, which denotes the labels of the training data. Hence, we have $\alpha \in \mathbb{R}^{n \times n_{\mathrm{cls}}}$. During the query, for each query $x \in \mathbb{R}^d$, we will have a prediction $p_{\mathrm{pred}} \in \mathbb{R}^{n_{\mathrm{cls}}}$ by the NTK regression. Then, we apply argmax to p_{pred}, and we will get the final predicted label of the query x. We randomly choose $1,000$ images for training and 100 for testing for each class. Namely, we will have $10,000$ in training images and $1,000$ in test images.

Feature Extraction. CIFAR-10 images possess a high-dimensional nature ($32 \times 32 \times 3 = 3,072$ dimensions), which poses a challenge for NTK Regression. To address this, we leverage the power of ResNet [59] to reduce the dimension. Following the approach of [9], we employ ResNet-18 to encode the images and extract features from the network's last layer, yielding a 512-dimensional feature representation for each image.

Feature Normalization. Prior to training our NTK Regression, we normalize all image features such that each feature vector's $\mathcal{L}_2$ norm is equal to 1.

NTK Regression Setup. For both NTK Regression and the NTK Regression Kernel Matrix, we select $m = 256$ neurons and a random Gaussian variance of $\sigma = 1$. This means that for each $r \in [m]$, the weights w_r are drawn from the normal distribution $\mathcal{N}(0, I_{d \times d})$. Additionally, we set the regularization factor $\lambda = 10$.

6.2. Experiment Results Analysis

Following the experimental setup detailed in Section 6.1, we present the results in Fig. 1.

During the execution of NTK Regression, we initially compute H^{dis} (as defined in Definition 3.5) based on the quadratic activation between the training data and m neurons. As H^{dis} is a symmetric matrix, it is also positive semi-definite. Then, in accordance with the notation in Definition 3.8, we define $K = H^{\mathrm{dis}}$. We then apply the Gaussian Sampling Mechanism, as described in Section 4, to privatize K, denoting the private version as $\widetilde{K}$. Due to the properties of the Gaussian Sampling Mechanism, $\widetilde{K}$ remains symmetric and thus positive semi-definite. Lemma 5.4 guarantees that $\widetilde{K}$ is $(\epsilon_\alpha, \delta_\alpha)$-DP.

We then compute the private α, denoted as $\widetilde{\alpha}$, by $\widetilde{\alpha} = (\widetilde{K} + \lambda I_{n \times n})^{-1} Y$. By the Post-processing Lemma of differential privacy (refer to Lemma 3.3), we confirm that $\widetilde{\alpha}$ is also $(\epsilon_\alpha, \delta_\alpha)$-DP.

Subsequently, we privatize the kernel function $\mathsf{K}(x, X)$. As described in Section 5.4, we apply truncated Laplace noise on X to get the private version $\widetilde{X}$, which is (ϵ_X, δ_X)-DP. Then, by the post-processing lemma, for any query $x \in \mathbb{R}^d$, we have $\mathsf{K}(x, \widetilde{X})$ is (ϵ_X, δ_X)-DP.

Consequently, applying the composition lemma, we can have the private NTK regression is (ϵ, δ)-DP.

In our experiment, we fix the differential privacy parameter $\delta = 2 \times 10^{-3}$. We recall that Δ is defined in Definition 5.2, and M is defined in Definition 5.1. To satisfy **Condition 5** in Condition 5.3, we must ensure $M < \Delta$.

We select k to be greater than or equal to $8 \cdot \log(1/\delta)$, ensuring that for any $\epsilon > 0$, the condition $\epsilon / \sqrt{8k \log(1/\delta)} \leq \epsilon / (8 \log(1/\delta))$ holds. Consequently, we have $\Delta = \epsilon / \sqrt{8k \log(1/\delta)}$ (see also Definition 5.2).

Under this setup, the condition $M < \Delta$ is equivalent to:

$$k \leq \frac{\epsilon^2 \eta_{\min}^2}{8 \log(1/\delta) n^2 \sigma^4 B^8 \beta^2} \tag{1}$$

In our experimental setup, we have $\sigma = 1$, $B = 1$, $\eta_{\min} = 7 \times 10^{-3}$, and $n = 10^3$. We assume $\beta = 10^{-6}$. We then compute the upper bound for k using Eq. (1) and adhere to this upper bound when conducting our experiments. The outcomes are presented in Fig. 1.

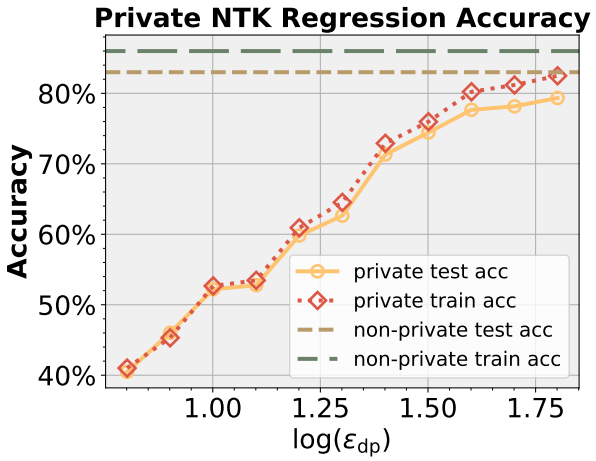

Figure 1. The trade-off between the accuracy parameter and privacy parameter. We conduct experiments on different privacy budget ϵ, where we fixed the $\delta = 2 \times 10^{-3}$, and we assume that $\beta = 10^{-6}$ in our experiments. The x-axis denotes the $\log(\epsilon_{\text{dp}})$, where the log denotes $\log_{10}$. The y-axis denotes the binary classification accuracy. As privacy budget ϵ_{dp} increase, both private test acc and private train acc approach to non-private train acc and non-private test acc, respectively.

7. Discussion

DP in kernel and gradient. In DP-SGD [1], they add Gaussian noise on the gradient for privacy. As they are a first-order algorithm, their function sensitivity is more robust for single-step training. However, as discussed in Section 1, to guarantee DP for the whole training process, their DP Gaussian noise variance will increase as T becomes larger (see Theorem 1 in [1]). The DP-SGD is not practical when T is too large. On the other hand, our NTK regression setting is a second-order algorithm involving kernel matrix inverse. Then, our key technical issues are (1) introducing a PSD noise matrix to keep kernel PSD property and (2) using L_2 regularization to make the kernel sensitivity more robust (see more details in Section 5).

Where to add noise? In the work, we add noise both on the kernel function $\mathsf{K}(x, X)$ and the the α to make the entire NTK regression private.

Others may argue that if we can only add noise on $\mathsf{K}(x, X)$ or α to ensure the privacy of the NTK regression. However, we argue that this is not feasible. The reasons are as follows. The primary reason is that we need to apply DP's post-processing lemma to ensure NTK regression's privacy. Therefore, we need to ensure the privacy of all the inputs we cared of the NTK regression. Since the NTK regression can be viewed as a function $\mathsf{F}(X, Y, K)$, which takes X, Y and K (the NTK matrix) as the inputs. Since we only aim to protect the sensitive information in X, we can view the NTK regression as a function $\mathsf{F}(X, K)$ only takes X and K as the inputs. Hence, we need to ensure privacy both on $\mathsf{K}(x, X)$ (corresponds to the input X) and α (corresponds to the input K) to have the privacy guarantees for the entire NTK regression.

Why NTK rather than Neural Networks (NNs)? We elucidate our preference for NTK-regression over traditional NNs based on two primary aspects. (1) Traditional NNs present analytical challenges. (2) NTK-regression effectively emulates the training dynamics of overparameterized NNs, facilitating a more tractable analysis.

To begin with, the analysis of traditional NNs is far from straightforward. Most modern NNs incorporate a variety of non-linear activation functions, complicating the derivation of theoretical bounds such as sensitivity and utility bounds. Consequently, the simplistic bounds for NNs are often impractically loose. Additionally, the intricacies of the training process for NNs, which lacks a closed-form solution or guaranteed global optimality in stochastic gradient descent (SGD), render a thorough analysis exceptionally difficult.

In contrast, the analysis of NTK regression serves as an good starting point. The NTK captures the essence of overparameterized NNs' behavior. The kernel function's and the linear properties in NTK regression allow for the derivation of closed-form solutions for its constituent parts, significantly simplifying analysis.

Consequently, this study adopts NTK regression as its analytical foundation, deferring a detailed examination of traditional NNs to our future research.

8. Conclusion

In conclusion, we have presented the first DP guarantees for NTK regression. From the theoretical side, we provide differential privacy guarantees for the NTK regression, and the theoretical utility bound for the private NTK regression. From the experimental side, we conduct validation experiments on the ten-class classification task on the CIFAR-10 dataset, which demonstrates our algorithm preserves good utility under a small private budget. This work opens new avenues for privacy-preserving deep learning using an NTK-based algorithm.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016. 1, 2, 8

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[3] Daniel Alabi, Pravesh K Kothari, Pranay Tankala, Prayaag Venkat, and Fred Zhang. Privately estimating a gaussian: Efficient, robust, and optimal. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 483–496, 2023. 5, 20

[4] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *Advances in neural information processing systems*, 32, 2019. 2

[5] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, 2019. 2

[6] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. On the convergence rate of training recurrent neural networks. In *NeurIPS*, 2019. 2

[7] Josh Alman, Jiehao Liang, Zhao Song, Ruizhe Zhang, and Danyang Zhuo. Bypass exponential time preprocessing: Fast neural network training via weight-data correlation preprocessing. In *NeurIPS*, 2023. 2

[8] Alexandr Andoni, Piotr Indyk, Sepideh Mahabadi, and Shyam Narayanan. Differentially private approximate near neighbor counting in high dimensions. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 43544–43562, 2023. 2, 3

[9] Josh Arnold. Resnet-extract-image-feature-pytorch-python. https://github.com/josharnoldjosh/Resnet-Extract-Image-Feature-Pytorch-Python, 2018. 7

[10] Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, pages 322–332. PMLR, 2019. 2

[11] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. *Advances in neural information processing systems*, 32, 2019. 1

[12] Martin Aumüller, Anders Bourgeat, and Jana Schmurr. Differentially private sketches for jaccard similarity estimation. In *Similarity Search and Applications: 13th International Conference, SISAP 2020, Copenhagen, Denmark, September 30–October 2, 2020, Proceedings 13*, pages 18–32. Springer, 2020. 2

[13] Arturs Backurs, Zinan Lin, Sepideh Mahabadi, Sandeep Silwal, and Jakub Tarnawski. Efficiently computing similarities to private datasets. *arXiv preprint arXiv:2403.08917*, 2024. 2

[14] Rouzbeh Behnia, Mohammadreza Reza Ebrahimi, Jason Pacheco, and Balaji Padmanabhan. 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, 2022. 2

[15] Sergei Bernstein. On a modification of chebyshev's inequality and of the error formula of laplace. *Ann. Sci. Inst. Sav. Ukraine, Sect. Math*, 1(4):38–49, 1924. 17

[16] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021. 1

[17] Jan van den Brand, Binghui Peng, Zhao Song, and Omri Weinstein. Training (overparametrized) neural networks in near-linear time. In *ITCS*, 2021. 2

[18] Yuan Cao, Zixiang Chen, Misha Belkin, and Quanquan Gu. Benign overfitting in two-layer convolutional neural networks. *Advances in neural information processing systems*, 35:25237–25250, 2022. 2

[19] Yuan Cao, Zhiying Fang, Yue Wu, Ding-Xuan Zhou, and Quanquan Gu. Towards understanding the spectral bias of deep learning, 2020. 2

[20] Yuan Cao and Quanquan Gu. Generalization bounds of stochastic gradient descent for wide and deep neural networks. *Advances in neural information processing systems*, 32, 2019. 2

[21] Bo Chen, Yingyu Liang, Zhizhou Sha, Zhenmei Shi, and Zhao Song. Hsr-enhanced sparse attention acceleration, 2024. 2

[22] Justin Y Chen, Shyam Narayanan, and Yinzhan Xu. All-pairs shortest path distances with differential privacy: Improved algorithms for bounded and unbounded weights. *arXiv preprint arXiv:2204.02335*, 2022. 2

[23] Zixiang Chen, Yuan Cao, Difan Zou, and Quanquan Gu. How much over-parameterization is sufficient to learn deep relu networks? *arXiv preprint arXiv:1911.12360*, 2019. 2

[24] Yeshwanth Cherapanamjeri, Sandeep Silwal, David P Woodruff, Fred Zhang, Qiuyi Zhang, and Samson Zhou. Robust algorithms on adaptive inputs from bounded adversaries. *arXiv preprint arXiv:2304.07413*, 2023. 2

[25] Lenaic Chizat and Francis Bach. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on learning theory*, pages 1305–1338. PMLR, 2020. 2

[26] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, 2019. 2

[27] Timothy Chu, Zhao Song, and Chiwun Yang. How to protect copyright data in optimization of large language models? *arXiv preprint arXiv:2308.12247*, 2023. 2

[28] Vincent Cohen-Addad, Alessandro Epasto, Vahab Mirrokni, Shyam Narayanan, and Peilin Zhong. Near-optimal private and scalable k-clustering. *Advances in Neural Information Processing Systems*, 35:10462–10475, 2022. 2

[29] Vincent Cohen-Addad, Chenglin Fan, Silvio Lattanzi, Slobodan Mitrovic, Ashkan Norouzi-Fard, Nikos Parotsidis, and Jakub M Tarnawski. Near-optimal correlation clustering with privacy. *Advances in Neural Information Processing Systems*, 35:33702–33715, 2022. 2

[30] Itai Dinur, Uri Stemmer, David P Woodruff, and Samson Zhou. On differential privacy and adaptive data analysis with bounded space. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 35–65. Springer, 2023. 2

[31] Wei Dong, Zijun Chen, Qiyao Luo, Elaine Shi, and Ke Yi. Continual observation of joins under differential privacy. *Proceedings of the ACM on Management of Data*, 2(3):1–27, 2024. 2

[32] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International conference on machine learning*, pages 1675–1685. PMLR, 2019. 2

[33] Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In *ICLR*. arXiv preprint arXiv:1810.02054, 2019. 2

[34] Cynthia Dwork. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, pages 1–19. Springer, 2008. 2

[35] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pages 265–284. Springer, 2006. 2

[36] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014. 1, 2, 3

[37] Kennedy Edemacu and Xintao Wu. Privacy preserving prompt engineering: A survey. *arXiv preprint arXiv:2404.06001*, 2024. 2

[38] Marek Eliáš, Michael Kapralov, Janardhan Kulkarni, and Yin Tat Lee. Differentially private release of synthetic graphs. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 560–578. SIAM, 2020. 2

[39] Alessandro Epasto, Vahab Mirrokni, Shyam Narayanan, and Peilin Zhong. k-means clustering with distance-based privacy. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[40] Hossein Esfandiari, Vahab Mirrokni, and Shyam Narayanan. Tight and robust private mean estimation with few users. In *International Conference on Machine Learning*, pages 16383–16412. PMLR, 2022. 2

[41] Chenglin Fan and Ping Li. Distances release with differential privacy in tree and grid graph. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 2190–2195. IEEE, 2022. 2

[42] Chenglin Fan, Ping Li, and Xiaoyun Li. k-median clustering via metric embedding: towards better initialization with differential privacy. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[43] Alireza Farhadi, MohammadTaghi Hajiaghayi, and Elaine Shi. Differentially private densest subgraph. In *International Conference on Artificial Intelligence and Statistics*, pages 11581–11597. PMLR, 2022. 2

[44] Yeqi Gao, Sridhar Mahadevan, and Zhao Song. An over-parameterized exponential regression. *arXiv preprint arXiv:2303.16504*, 2023. 2

[45] Yeqi Gao, Lianke Qin, Zhao Song, and Yitan Wang. A sublinear adversarial training algorithm. In *ICLR*, 2024. 2

[46] Yeqi Gao, Zhao Song, Xin Yang, and Yufa Zhou. Differentially private attention computation. In *Neurips Safe Generative AI Workshop 2024*, 2024. 1, 2, 4, 5, 19, 20

[47] Mario Geiger, Stefano Spigler, Arthur Jacot, and Matthieu Wyart. Disentangling feature and lazy training in deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, page 113301, 2020. 2

[48] Quan Geng, Wei Ding, Ruiqi Guo, and Sanjiv Kumar. Tight analysis of privacy and utility tradeoff in approximate differential privacy. In *International Conference on Artificial Intelligence and Statistics*, pages 89–99. PMLR, 2020. 2, 3

[49] Badih Ghazi, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, and Kewen Wu. On differentially private counting on trees. In *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261, page 66. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2023. 2

[50] Aditya Golatkar, Alessandro Achille, Yu-Xiang Wang, Aaron Roth, Michael Kearns, and Stefano Soatto. Mixed differential privacy in computer vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8376–8386, 2022. 2

[51] Sivakanth Gopi, Yin Tat Lee, and Daogao Liu. Private convex optimization via exponential mechanism. In *Conference on Learning Theory*, pages 1948–1989. PMLR, 2022. 2

[52] Sivakanth Gopi, Yin Tat Lee, Daogao Liu, Ruoqi Shen, and Kevin Tian. Private convex optimization in general norms. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 5068–5089. SIAM, 2023. 2

[53] Sivakanth Gopi, Yin Tat Lee, and Lukas Wutschitz. Numerical composition of differential privacy. *Advances in Neural Information Processing Systems*, 34:11631–11642, 2021. 2

[54] Jiuxiang Gu, Chenyang Li, Yingyu Liang, Zhenmei Shi, and Zhao Song. Exploring the frontiers of softmax: Provable optimization, applications in diffusion model, and beyond. *arXiv preprint arXiv:2405.03251*, 2024. 2

[55] Jiuxiang Gu, Chenyang Li, Yingyu Liang, Zhenmei Shi, Zhao Song, and Tianyi Zhou. Fourier circuits in neural networks: Unlocking the potential of large language models in mathematical reasoning and modular arithmetic. *arXiv preprint arXiv:2402.09469*, 2024. 2

[56] Jiuxiang Gu, Yingyu Liang, Zhenmei Shi, Zhao Song, and Chiwun Yang. Toward infinite-long prefix in transformer. *arXiv preprint arXiv:2406.14036*, 2024. 2

[57] Jiuxiang Gu, Yingyu Liang, Zhenmei Shi, Zhao Song, and Yufa Zhou. Unraveling the smoothness properties of diffusion models: A gaussian mixture perspective. *arXiv preprint arXiv:2405.16418*, 2024. 2

[58] Yinbin Han, Meisam Razaviyayn, and Renyuan Xu. Neural network-based score estimation in diffusion models: Optimization and generalization. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[59] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7

[60] Samuel B Hopkins, Gautam Kamath, Mahbod Majid, and Shyam Narayanan. Robustness implies privacy in statistical estimation. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 497–506, 2023. 2

[61] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021. 2

[62] Hang Hu, Zhao Song, Omri Weinstein, and Danyang Zhuo. Training overparametrized neural networks in sublinear time. *arXiv preprint arXiv:2208.04508*, 2022. 2

[63] Ziyue Huang and Ke Yi. Approximate range counting under differential privacy. In *37th International Symposium on Computational Geometry (SoCG 2021)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2021. 2

[64] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018. 1, 2

[65] Ziwei Ji and Matus Telgarsky. Polylogarithmic width suffices for gradient descent to achieve arbitrarily small test error with shallow relu networks. In *International Conference on Learning Representations*, 2019. 2

[66] Haibo Jin, Leyang Hu, Xinuo Li, Peiyan Zhang, Chonghan Chen, Jun Zhuang, and Haohan Wang. Jailbreakzoo: Survey, landscapes, and horizons in jailbreaking large language and vision-language models. *arXiv preprint arXiv:2407.01599*, 2024. 1

[67] Christopher Jung, Katrina Ligett, Seth Neel, Aaron Roth, Saeed Sharifi-Malvajerdi, and Moshe Shenfeld. A new analysis of differential privacy's generalization guarantees. *arXiv preprint arXiv:1909.03577*, 2019. 2

[68] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 7

[69] Beatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, pages 1302–1338, 2000. 17

[70] Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. *Advances in neural information processing systems*, 2019. 2

[71] Jason D Lee, Ruoqi Shen, Zhao Song, Mengdi Wang, et al. Generalized leverage score sampling for neural networks. *Advances in Neural Information Processing Systems*, 33:10775–10787, 2020. 2, 3

[72] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020. 2

[73] Haoran Li, Yulin Chen, Jinglong Luo, Yan Kang, Xiaojin Zhang, Qi Hu, Chunkit Chan, and Yangqiu Song. Privacy in large language models: Attacks, defenses and future directions. *arXiv preprint arXiv:2310.10383*, 2023. 1

[74] Haoran Li, Dadi Guo, Wei Fan, Mingshi Xu, Jie Huang, Fanpu Meng, and Yangqiu Song. Multi-step jailbreaking privacy attacks on chatgpt. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4138–4153, 2023. 1

[75] Ninghui Li, Min Lyu, Dong Su, and Weining Yang. *Differential privacy: From theory to practice*. Springer, 2017. 2

[76] Ping Li and Xiaoyun Li. Differential privacy with random projections and sign random projections. *arXiv preprint arXiv:2306.01751*, 2023. 2

[77] Ping Li and Xiaoyun Li. Smooth flipping probability for differential private sign random projection methods. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[78] Xiaoxiao Li, Zhao Song, and Jiaming Yang. Federated adversarial learning: A framework with convergence analysis. In *International Conference on Machine Learning*, pages 19932–19959. PMLR, 2023. 2

[79] Xiaoyu Li, Yingyu Liang, Zhenmei Shi, Zhao Song, and Junwei Yu. Fast john ellipsoid computation with differential privacy optimization. *arXiv preprint arXiv:2408.06395*, 2024. 2

[80] Xiaoyun Li and Ping Li. Differentially private one permutation hashing and bin-wise consistent weighted sampling. *arXiv preprint arXiv:2306.07674*, 2023. 2

[81] Xuechen Li, Daogao Liu, Tatsunori B Hashimoto, Huseyin A Inan, Janardhan Kulkarni, Yin-Tat Lee, and Abhradeep Guha Thakurta. When does differentially private learning not suffer in high dimensions? *Advances in Neural Information Processing Systems*, 35:28616–28630, 2022. 2

[82] Xuechen Li, Florian Tramer, Percy Liang, and Tatsunori Hashimoto. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*, 2021. 2

[83] Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. *Advances in neural information processing systems*, 31, 2018. 2

[84] Yingyu Liang, Zhizhou Sha, Zhenmei Shi, Zhao Song, and Yufa Zhou. Looped relu mlps may be all you need as practical programmable computers, 2024. 2

[85] Yingyu Liang, Zhizhou Sha, Zhenmei Shi, Zhao Song, and Yufa Zhou. Multi-layer transformers gradient can be approximated in almost linear time. *arXiv preprint arXiv:2408.13233*, 2024. 2

[86] Yingyu Liang, Zhenmei Shi, Zhao Song, and Yufa Zhou. Differential privacy of cross-attention with provable guarantee. *arXiv preprint arXiv:2407.14717*, 2024. 2

[87] Bo Liu, Ming Ding, Sina Shaham, Wenny Rahayu, Farhad Farokhi, and Zihuai Lin. When machine learning meets privacy: A survey and outlook. *ACM Computing Surveys (CSUR)*, 54(2):1–36, 2021. 2

[88] Erzhi Liu, Jerry Yao-Chieh Hu, Alex Reneau, Zhao Song, and Han Liu. Differentially private kernel density estimation. *arXiv preprint arXiv:2409.01688*, 2024. 2

[89] Zhihao Liu, Jian Lou, Wenjie Bao, Zhan Qin, and Kui Ren. Differentially private zeroth-order methods for scalable large language model finetuning. *arXiv preprint arXiv:2402.07818*, 2024. 2

[90] Kaifeng Lyu, Zhiyuan Li, Runzhe Wang, and Sanjeev Arora. Gradient descent on two-layer nets: Margin maximization and simplicity bias. *Advances in Neural Information Processing Systems*, 34:12978–12991, 2021. 2

[91] Sadhika Malladi, Alexander Wettig, Dingli Yu, Danqi Chen, and Sanjeev Arora. A kernel-based view of language model fine-tuning. In *International Conference on Machine Learning*, pages 23610–23641. PMLR, 2023. 2

[92] Justus Mattern, Zhijing Jin, Benjamin Weggenmann, Bernhard Schölkopf, and Mrinmaya Sachan. Differentially private language models for secure data sharing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4860–4873. Association for Computational Linguistics, 2022. 2

[93] Alexander G de G Matthews, Mark Rowland, Jiri Hron, Richard E Turner, and Zoubin Ghahramani. Gaussian process behaviour in wide deep neural networks. In *International Conference on Learning Representations*, 2018. 2

[94] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit. In *Conference on Learning Theory*, pages 2388–2464. PMLR, 2019. 2

[95] Lingsheng Meng and Bing Zheng. The optimal perturbation bounds of the moore–penrose inverse under the frobenius norm. *Linear algebra and its applications*, 432(4):956–963, 2010. 39

[96] Andrea Montanari and Yiqiao Zhong. The interpolation phase transition in neural networks: Memorization and generalization under lazy training. *The Annals of Statistics*, 50(5):2816–2847, 2022. 2

[97] Alexander Munteanu, Simon Omlor, Zhao Song, and David Woodruff. Bounding the width of neural networks via coupled initialization a worst case analysis. In *International Conference on Machine Learning*, pages 16083–16122. PMLR, 2022. 2

[98] Shyam Narayanan. Private high-dimensional hypothesis testing. In *Conference on Learning Theory*, pages 3979–4027. PMLR, 2022. 2

[99] Shyam Narayanan. Better and simpler lower bounds for differentially private statistical estimation. *arXiv preprint arXiv:2310.06289*, 2023. 2

[100] Roman Novak, Lechao Xiao, Jaehoon Lee, Yasaman Bahri, Daniel A Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Bayesian convolutional neural networks with many channels are gaussian processes. In *International Conference on Learning Representations*, 2019. 2

[101] Samet Oymak, Zalan Fabian, Mingchen Li, and Mahdi Soltanolkotabi. Generalization guarantees for neural networks via harnessing the low-rank structure of the jacobian. *arXiv preprint arXiv:1906.05392*, 2019. 2

[102] Samet Oymak and Mahdi Soltanolkotabi. Overparameterized nonlinear learning: Gradient descent takes the shortest path? In *International Conference on Machine Learning*, pages 4951–4960. PMLR, 2019. 2

[103] Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020. 2

[104] Omkar Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *BMVC 2015-Proceedings of the British Machine Vision Conference 2015*. British Machine Vision Association, 2015. 1

[105] Natalia Ponomareva, Hussein Hazimeh, Alex Kurakin, Zheng Xu, Carson Denison, H Brendan McMahan, Sergei Vassilvitskii, Steve Chien, and Abhradeep Guha Thakurta. How to dp-fy ml: A practical guide to machine learning with differential privacy. *Journal of Artificial Intelligence Research*, 77:1113–1201, 2023. 2

[106] Lianke Qin, Zhao Song, and Baocheng Sun. Is solving graph neural tangent kernel equivalent to training graph neural network? *arXiv preprint arXiv:2309.07452*, 2023. 2

[107] Benjamin IP Rubinstein, Peter L Bartlett, Ling Huang, and Nina Taft. Learning in a large function space: Privacy-preserving mechanisms for svm learning. *arXiv preprint arXiv:0911.5708*, 2009. 1

[108] Weiyan Shi, Ryan Shea, Si Chen, Chiyuan Zhang, Ruoxi Jia, and Zhou Yu. 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, 2022. 2

[109] Zhenmei Shi, Jiefeng Chen, Kunyang Li, Jayaram Raghuram, Xi Wu, Yingyu Liang, and Somesh Jha. The trade-off between universality and label efficiency of representations from contrastive learning. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[110] Zhenmei Shi, Yifei Ming, Ying Fan, Frederic Sala, and Yingyu Liang. Domain generalization via nuclear norm regularization. In *Conference on Parsimony and Learning (Proceedings Track)*, 2023. 2

[111] Zhenmei Shi, Junyi Wei, and Yingyu Liang. A theoretical analysis on feature learning in neural networks: Emergence from inputs and advantage over fixed features. In *International Conference on Learning Representations*, 2021. 2

[112] Zhenmei Shi, Junyi Wei, and Yingyu Liang. Provable guarantees for neural networks via gradient feature learning. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[113] Tanmay Singh, Harshvardhan Aditya, Vijay K Madisetti, and Arshdeep Bahga. Whispered tuning: Data privacy preservation in fine-tuning llms through differential privacy. *Journal of Software Engineering and Applications*, 17(1):1–22, 2024. 2

[114] Zhao Song, Yitan Wang, Zheng Yu, and Lichen Zhang. Sketching for first order method: efficient algorithm for low-bandwidth channel and vulnerability. In *International Conference on Machine Learning*, pages 32365–32417. PMLR, 2023. 2

[115] Zhao Song, Shuo Yang, and Ruizhe Zhang. Does preprocessing help training over-parameterized neural networks? *Advances in Neural Information Processing Systems*, 34:22890–22904, 2021. 2

[116] Zhao Song and Xin Yang. Quadratic suffices for over-parametrization via matrix chernoff bound. *arXiv preprint arXiv:1906.03593*, 2019. 2

[117] Zhao Song, Xin Yang, Yuanyuan Yang, and Lichen Zhang. Sketching meets differential privacy: fast algorithm for dynamic kronecker projection maintenance. In *International Conference on Machine Learning (ICML)*, pages 32418–32462. PMLR, 2023. 2

[118] Zhao Song and Mingquan Ye. Efficient asynchronize stochastic gradient algorithm with structured data. *arXiv preprint arXiv:2305.08001*, 2023. 2

[119] Zhao Song, Lichen Zhang, and Ruizhe Zhang. Training multi-layer over-parametrized neural network in subquadratic time. In *ITCS*, 2024. 2

[120] Thomas Steinke. Composition of differential privacy & privacy amplification by subsampling. *arXiv preprint arXiv:2210.00597*, 2022. 2

[121] Yiyou Sun, Zhenmei Shi, and Yixuan Li. A graph-theoretic framework for understanding open-world semi-supervised learning. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[122] Yiyou Sun, Zhenmei Shi, Yingyu Liang, and Yixuan Li. When and how does known class help discover unknown ones? provable understanding through spectral analysis. In *International Conference on Machine Learning*, pages 33014–33043. PMLR, 2023. 2

[123] Yilin Wang, Zeyuan Chen, Liangjun Zhong, Zheng Ding, Zhizhou Sha, and Zhuowen Tu. Dolfin: Diffusion layout transformers without autoencoder. *arXiv preprint arXiv:2310.16305*, 2023. 2

[124] Yilin Wang, Haiyang Xu, Xiang Zhang, Zeyuan Chen, Zhizhou Sha, Zirui Wang, and Zhuowen Tu.

Omnicontrolnet: Dual-stage integration for conditional image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7436–7448, 2024. 2

[125] Yuqing Wang, Ye He, and Molei Tao. Evaluating the design space of diffusion-based generative models. *arXiv preprint arXiv:2406.12839*, 2024. 2

[126] Zirui Wang, Zhizhou Sha, Zheng Ding, Yilin Wang, and Zhuowen Tu. Tokencompose: Grounding diffusion with token-level supervision. *arXiv preprint arXiv:2312.03626*, 2023. 2

[127] Per-Åke Wedin. Perturbation theory for pseudo-inverses. *BIT Numerical Mathematics*, 13:217–232, 1973. 39

[128] Rongzhe Wei, Eleonora Kreačić, Haoyu Wang, Haoteng Yin, Eli Chien, Vamsi K Potluru, and Pan Li. On the inherent privacy properties of discrete denoising diffusion models. *arXiv preprint arXiv:2310.15524*, 2023. 2

[129] David Woodruff, Fred Zhang, and Samson Zhou. On robust streaming for learning with experts: algorithms and lower bounds. *Advances in Neural Information Processing Systems*, 36:79518–79539, 2023. 2

[130] Zhuoyan Xu, Zhenmei Shi, Junyi Wei, Fangzhou Mu, Yin Li, and Yingyu Liang. Towards few-shot adaptation of foundation models via multitask finetuning. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[131] Greg Yang. Scaling limits of wide neural networks with weight sharing: Gaussian process behavior, gradient independence, and neural tangent kernel derivation. *arXiv preprint arXiv:1902.04760*, 2019. 2

[132] Mengmeng Yang, Taolin Guo, Tianqing Zhu, Ivan Tjuawinata, Jun Zhao, and Kwok-Yan Lam. Local differential privacy and its applications: A comprehensive survey. *Computer Standards & Interfaces*, page 103827, 2023. 2

[133] Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, et al. Differentially private fine-tuning of language models. In *International Conference on Learning Representations*, 2021. 2

[134] Jiahao Yu, Haozheng Luo, Jerry Yao-Chieh Hu, Wenbo Guo, Han Liu, and Xinyu Xing. Enhancing jailbreak attack against large language models through silent tokens. *arXiv preprint arXiv:2405.20653*, 2024. 1

[135] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM computing surveys (CSUR)*, 52(1):1–38, 2019. 1

[136] Ying Zhao and Jinjun Chen. A survey on differential privacy for unstructured data content. *ACM Computing Surveys (CSUR)*, 54(10s):1–28, 2022. 2

[137] Chunyan Zheng, Keke Sun, Wenhao Zhao, Haibo Zhou, Lixing Jiang, Shaoyang Song, and Chunlai Zhou. Locally differentially private in-context learning. In *LREC/COLING*, 2024. 2

[138] Mo Zhou, Rong Ge, and Chi Jin. A local convergence theory for mildly over-parameterized two-layer neural network. In *Conference on Learning Theory*, pages 4577–4632. PMLR, 2021. 2

[139] Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep relu networks. *arXiv preprint arXiv:1811.08888*, 2018. 2

[140] Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Gradient descent optimizes over-parameterized deep relu networks. *Machine learning*, 109:467–492, 2020. 2

[141] Difan Zou and Quanquan Gu. An improved analysis of training over-parameterized deep neural networks. *Advances in neural information processing systems*, 32, 2019. 2