Location and Data Privacy Preservation in Intelligent Systems

by

Xinyue Zhang

A dissertation submitted to the Department of Electrical and Computer Engineering,

Cullen College of Engineering

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in Electrical Engineering

Chair of Committee: Miao Pan

Committee Member: Zhu Han

Committee Member: Hien Van Nguyen

Committee Member: Xin Fu

Committee Member: Ming Li

University of Houston
May 2021

ACKNOWLEDGMENTS

First and foremost, I would like to express my deepest gratitude to my advisor Dr. Miao Pan for his invaluable advice, continuous support, and patience during my Ph.D. study. His immense knowledge, keen insight and plentiful experience has sharpen my thinking and encouraged me in all the time of my academic research and daily life. It was a great privilege and honor to work and study under his guidance.

Morevoer, I would also like to extend my sincere appreciation to my committee, Dr. Zhu Han, Dr. Hien Van Nguyen, Dr. Xin Fu and Dr. Ming Li, who provided ingenious suggestions and insightful feedback on my research. The completion of my dissertation would not have been possible without the support and nurturing of them. I am also grateful to all other scholars I have collaborated with, Dr. Pan Li, Dr. Yuanxiong Guo, Dr. Stephen TC Wong, Dr. Minglei Shu, Dr. Lixin Li, Dr. Xuanheng Li, Dr. Yong Li, Dr. Riku Jäntti, Dr. Yinglong Wang, Dr. Hongning Li, Dr. Qingqi Pei, Dr. Xiaoxia Huang, Dr. Jie Wang, and Dr. Tingting Yang. I really appreciate their constructive critique and helpful comments during every discussion.

Special thanks to all my dear friends and colleagues, Dr. Jingyi Wang, Dr. Sai Mounika Errapotu, Debing Wei, Jiahao Ding, Dian Shi, Pavana Prakash, Rui Chen, Chenpei Huang, Dr. Feng Hu, Liang Li, Maoqiang Wu, Pengyu Yuan, Steban Soto, Yu Li, and many others. It is their kind help and support that have made my study and life in Houston a wonderful time. I also want to thank and recognize the dedication of the exceptional staff across the university whose timely assistance cannot be overestimated.

I'm deeply indebted to my parents and my husband for their wise counsel and sympathetic ear throughout my life. Their profound belief in my abilities gives me strength to chase my dreams.

ABSTRACT

Due to the ubiquitous mobile devices with embedded sensors and connectivity over the internet, the Internet of things (IoT) has evolved. The IoT brings the explosive growth of devices connected and controlled by the Internet. The enormous collection of connected sensors and devices makes a significant contribution to the volume of data collected, which brings us to the big data era. Intelligent system (IS) becomes an emerging paradigm for integrating big data, analytics, privacy, and artificial intelligence. The IS is any formal or informal system to manage data gathering, to obtain and process the data, to interpret the data, and to provide reasoned judgments to decision makers as a basis for action. In order to keep up with the continuous influx of data, machine learning is one of the best solutions for big data analysis, which is fast evolving during the last decade. With the development of machine learning technologies, it plays a critical role in IS. The IS, which integrates computations, communications and decision making, interacts with humans through many new modalities. However, privacy is an essential concern in IS since a large volume of users' daily and sensitive data is used in constituting systems, and users become increasingly concerned about the compromise of their personal information. Therefore, it is necessary to develop innovative privacy preserving approaches to prevent users' confidential information from illegal revealing while efficiently utilizing massive data generated from users.

In fact, there are trade-offs between the effectiveness of privacy protection and the convenience of data collection, communications, and energy consumption, which need proper considerations in system designs. The objectives of this dissertation are to develop efficient and reliable data analysis methods in various IS applications and protect the data privacy against malicious attacks through a combination of theoretical, simulation, and experimental studies. Given the challenge of privacy preservation and reliable data analysis, this work endeavors to develop a series of privacy preserving data analytic and processing methodologies through machine learning, optimization and differential privacy; and focuses on effectively integrating the data analysis and data privacy preservation techniques to provide the most desirable solutions for the state-of-the-art IS with various application-specific requirements.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

With the rapid development of the intelligent system (IS), there is more and more data available for data analysis tasks. In order to glean useful insights from huge data sets, machine learning becomes one of the most promising solutions by unleashing the potential of the big data generated in the IS environments. In IS, healthcare systems have been a vital research area and deep learning holds great promise in improving healthcare and medicine. Medical institutions hold various modalities of medical data, such as electronic health records, biomedical images, and pathology test results. Based on medical data, deep neural network models are trained to address necessary healthcare concerns [1]. Examples include but not limited to: 1) deep learning models based on medical records outperformed traditional clinical models for detecting patterns in health trends and risk factors [2]; 2) deep learning model had high sensitivity and specificity for detecting diabetic retinopathy and macular edema in retinal fundus photographs [3]; 3) a mammography-based deep learning model was more accurate than traditional clinical models for predicting breast cancer risk [4].

When training a machine learning model, it is certain that training data with large size are yielding desirable results. Although huge amount of data is available and accessible in the big data era, in the medical and healthcare systems, there is still not enough data for training, thereby affecting the performance of the models. Generative adversarial network (GAN), first proposed in [5] is a potential solution that addresses the data scarcity problem in fields with limited data because of its ability to generate data similar to real data. Typically, GAN integrates deep learning and game theory to train two models, generator and discriminator, so that the generator can generate high fidelity fake samples from latent space, that are supposed to be indistinguishable from the real data samples. Despite the solid fundamental theories and impressive empirical experiments, training the original GAN is relatively tricky and improving the quality of generated images is still challenging. Deep convolutional GAN (DCGAN) proposed in [6] first attempted to deal with this problem by applying convolutional neural networks (CNNs) in GAN to enable stable training and improve the applicability of the generated images. GANs and its variants are being investigated to achieve effective results through data augmentation and overcome training data insufficiency for the models.

However, the application of deep learning on sensitive datasets is confronted with privacy threats. Especially, in the healthcare systems, the medical records contain personal private information like drug usage patterns of the individual patient. Medical institutions also hold patients' profile information such as home address, gender, age, etc. The private information might be unwittingly leaked when the aforementioned data is used for training a deep learning model [7, 8]. For example, attribute inference attacks [7] can utilize the trained model and incomplete information about a data point to infer the missing information for that point. The adversary could exploit such an attack to infer the target private information according to partial information on medical records. Another instance is model inversion attacks [9] that enable the image data used for classification inference to be recovered based on the intermediate output of convolutional neural networks (CNNs). The adversary could deploy model inversion attack to recover the medical images of any target patient and infer private health conditions accordingly. The risk of privacy leakage makes medical institutions increasingly less willing to share their data. This inevitably slows down the research progress at the intersection of deep learning and healthcare. Thus, it is necessary to evaluate the potential hazards of various attack models on medical data and develop corresponding defenses on such attacks.

The last decade has witnessed the exploding growth in the quantity and capability of consumer mobile devices such as smartphones, tablets, etc., and the proliferation of wireless services. IS is often distributed broadly across wide geographic areas and typically collect huge amounts of information for data analysis and decision making. Mobile IS that takes the advantages and extend the application domains of IS is a kind of foundational techniques to support the development of mobile and vehicular networking systems. The extended mobile IS includes mobile crowdsensing (MCS) and transportation network company (TNC) services.

The MCS has become a novel sensing and computing paradigm, due to the development of the GPS embedded mobile devices. The objective of MCS is to efficiently pair requested tasks and interested workers, involve workers to contribute and analyze data collected by their mobile devices, aggregate and utilize the reported data to reveal information for specific purposes [10]. In the MCS platform, there are plenty of candidate workers, who are waiting to accomplish tasks. When the task requesters send the task assignments to the MCS server, a subset of appropriate workers will be chosen to participate. Such procedure, called task allocation, is a significant problem that needs to be considered in MCS. Mostly, in task allocation, the workers' travel distance is referred as one of the most common utility metrics. On the workers' side, they prefer to participate in tasks located closer to themselves. On the MCS servers' side, longer travel distance means higher response latency and higher incentive bonus to the workers, which is not desirable [11]. Therefore, based on the travel distance, the tasks can be assigned to workers properly, hence enhancing the efficiency of the MCS.

The TNC service has become an indispensable option of people's daily commute. The TNC provide the ridesharing services to solve the dilemma of the firstmile/last-mile problem of public transit users and traffic congestion problem in urban areas. Such TNC services (e.g., Uber, Lyft, Didi, etc.) can pair TNC users and TNC vehicles according to their location information through the mobile apps in order to provide ridesharing [12]. With TNC services, the users are able to save time by changing parking to TNC user loading events. As TNC users can gain more time savings when using TNC services, it also provide more job opportunities for drivers. In [13], the authors note that ridesharing scheduled by TNC can occupy the capacity more efficiently than traditional taxicabs. In addition, due to the development of technologies of self-driving cars, Google autonomous driving projects Waymo has already been testing in Arizona. The reported status shows that TNC is supposed to launch the self-driving car services sooner or later. Therefore, it is necessary to have a new TNC vehicle scheduling scheme for both the ridesharing drivers and self-driving vehicles with high efficiency and flexibility.

Despite the popularity of MCS and TNC services, the prerequisites of workers' location sharing raise serious privacy concern. Data breaches could potentially happen in any part of the system, including the stages of data collection, data transmission, data operation, and data storage. Due to its importance and rapid advances in computing technologies, mobile IS and its communication networks inevitably become the targets of attackers and malicious parties. With the exposure of workers' location, the workers not only lose their privacy but also are vulnerable to various attacks, even some serious physical attacks. For instance, in August 2018, a violent robbery happened in Maryland. The victim is picked up by a fake Uber driver and the driver demanded the victim's wallet and cellphone with a handgun. Instead of robbery, more serious crimes targeting particular victims, such as kidnapping, sexual assault, etc., might happen because of the leakage of location information, since malicious party could learn people's life pattern through the disclosed locations.

To prevent the privacy leakage, in this dissertation, we leverage differential privacy [14], which is widely applied to many research areas. Intuitively, it works by injecting random noise to the data so that an adversary with arbitrary background knowledge cannot confidently make any conclusions about the raw data. Since the differentially private noises bring uncertainty to the data, in order to increase the data utility, we deploy several data-driven technology such as conditional value at risk (CVaR), adaptive method, and ζ-structure probability metric.

Specifically, we first illustrate the differential privacy preliminaries in Section 2. We evaluate the inference attack models for deep learning on medical data in Section 3. Then, in Section 4, we present an adaptive privacy preservation scheme on deep learning models. Basically, we inject the linear decaying differentially private noises into the gradient during each iteration to protect the training data while increase the utility of the model. In Section 5, we investigate privacy-preserving schemes in generative adversarial networks (GAN), where we add Laplace noises into the loss function of discriminator. As the generator is the post-processing procedure of the discriminator in GAN, the overall GAN is differentially private. In Section 6, we present task allocation optimization problem in mobile crowdsensing by considering the user's location privacy. We proposed two differentially-privacy based location protection shcemes, formulate the travel distance minimization problem, and utilize the CVaR to solve the optimization problem. In Section 7, we develop a data-driven method to schedule vehicles for the transportation network company with the considerations of the drivers' location privacy. At last, we present some possible future works to where privacy and secure technology can be applied in Section 8.

2 Differential Privacy Preliminaries

Differential privacy (DP) [14] that provides a strong standard privacy guarantee is being widely applied to many research areas. Basically, it is used to protect data providers' privacy when the statistical information of a database is publishing. Its wide acceptance is based on its merits of effectively protecting the data providers' privacy while publishing the statistical information of the databases. DP indicates that the participation of one patient in the training phase has an inconsiderable effect on the final deep network model. After conducting a randomized algorithm $\mathcal{M}$, the two probabilities, that from a query to the two neighbor databases D and $\hat{D}$ the returned values are the value o, is supposed to within a bound of ϵ given a probability $1-\delta$, where ϵ is the privacy confidence parameter and δ is the broken probability. The definition of *differential privacy* is shown as follows.

Definition 2.1 (**Differential Privacy**) A randomized algorithm $\mathcal{M}$ satisfies (ϵ, δ)-differential privacy if for any two adjacent datasets D and $\hat{D}$ that differ in only a single record, the absolute value of the privacy loss random variable of an output $o \in \mathit{Range}(\mathcal{M})$, $Z(o) = \log \frac{\Pr[\mathcal{M}(D)=o]}{\Pr[\mathcal{M}(\hat{D})=o]}$ is bounded by ϵ, with probability at least $1-\delta$.

The privacy budget ϵ controls the privacy preservation level and δ is the broken probability, and if $\delta = 0$, the randomized algorithm $\mathcal{M}$ is said to have ϵ-differential privacy. A larger ϵ means lower privacy level, and implies that there is a higher possibility to distinguish the outputs of the randomized algorithm $\mathcal{M}$ with two different input datasets. Intuitively, smaller ϵ means higher privacy preservation level. δ is the broken probability.

Normally, the standard approach to achieve ϵ-differential privacy is through Laplace mechanism defined as follows.

Definition 2.2 (**Laplace Mechanism**) Given a query function $Q : \mathcal{X} \to \mathcal{R}$, the Laplace mechanism that adds noise generated from the Laplace distribution $Lap(\frac{\Delta}{\epsilon})$ to the output of query function Q enjoys ϵ-differential privacy, where Δ is the l_1-sensitivity of the query Q, i.e., $\Delta = \sup_{\mathcal{X} \sim \mathcal{X}'} \|Q(\mathcal{X}) - Q(\mathcal{X}')\|_1$.

A generic method of achieving (ϵ, δ)-differential privacy is Gaussian mechanism [15] that adds Gaussian noise, calibrated to the query function's sensitivity, to the output. The sensitivity captures the maximum difference of the query function by a single record in the worst case. We define the sensitivity as follows.

Definition 2.3 (**l_2-Sensitivity**) The l_2 sensitivity of a query function $f(\cdot)$ that takes as input a dataset D is defined as

$$\Delta_f = \max_{D,\hat{D}} \|f(D) - f(\hat{D})\|_2, \tag{1}$$

where D and $\hat{D}$ are any two neighboring datasets differing in at most one record.

In this paper, we consider the gradient perturbation method to provide privacy guarantee of deep neural network. Thus, the query function f is the gradient of deep neural network. We can easily enforce a specific sensitivity value Δ_f by clipping the L_2-norm of gradient value. Based on the definition of sensitivity, we show the Gaussian mechanism in the following theorem.

Theorem 1 (**Gaussian Mechanism [15]**) *For a query function $f : \mathcal{D} \rightarrow \mathcal{R}^d$ with sensitivity Δ_f, the Gaussian Mechanism that adds noise generated from the Gaussian distribution $\mathcal{N}(0, \sigma^2\mathbb{I})$ to the output of f satisfies (ϵ, δ)-differential privacy, where $\epsilon, \delta \in (0, 1)$ and $\sigma \geq \frac{\sqrt{2\ln(1.25/\delta)}\Delta_f}{\epsilon}$.*

When applying gradient perturbation method in training phase of deep neural network, due to the large number of iterations, the composition property of differential privacy is important to estimate the privacy loss. Hence, we adopt truncated concentrated differential privacy (tCDP) [16], a new relaxation of differential privacy, to provide sharper and tighter analysis on the privacy loss for multiple iterative computations compared to (ϵ, δ)-DP. The definition of tCDP is defined as follows.

Definition 2.4 (**tCDP**) For all $\tau \in (1, \omega)$, a randomized algorithm $\mathcal{M}$ is (ρ, ω)-tCDP if for any neighboring datasets D and $\hat{D}$ and all $\alpha > 1$, we have

$$D_\tau(\mathcal{M}(x) || \mathcal{M}(x')) \leq \rho\alpha, \tag{2}$$

where $D_\tau(\cdot||\cdot)$ is the the Rényi divergence of order τ defined as follows.

Given two distributions μ and ν on a Banach space $(\mathcal{Z}, \|\cdot\|)$, here, we consider the Rényi divergence distance between them:

Definition 2.5 (Rényi Divergence [17]) Let $1 < \alpha < \infty$ and μ, ν be measures with $\mu \ll \nu$. The Rényi divergence of order α between μ and ν is defined as

$$D_\tau(\mu\|\nu) \doteq \frac{1}{\alpha - 1} \ln \int \left(\frac{\mu(z)}{\nu(z)} \right)^{\alpha} \nu(z) \,\mathrm{d}z. \tag{3}$$

Here we follow the convention that $\frac{0}{0} = 0$. If $\mu \not\ll \nu$, we define the Rényi divergence to be ∞. Rényi divergence of orders $\alpha = 1, \infty$ is defined by continuity.

In this paper, we mainly utilize the following properties of tCDP, shown in [16].

Lemma 1 *The Gaussian mechanism, in Theorem 1, satisfies $(\Delta_f^2/(2\sigma^2), \infty)$-tCDP.*

Lemma 2 *If randomized mechanisms $\mathcal{M}_1$ and $\mathcal{M}_2$ satisfy (ρ_1, ω_1)-tCDP, and (ρ_2, ω_2)-tCDP, their composition defined as $(\mathcal{M}_1, \mathcal{M}_2)$ is $(\rho_1 + \rho_2, \min(\omega_1, \omega_2))$-tCDP.*

Lemma 3 *If a randomized mechanism $\mathcal{M}$ satisfies (ρ, ω)-tCDP, then for any $\delta \geq 1/\exp((\omega - 1)^2\rho)$, $\mathcal{M}$ satisfies $(\rho + 2\sqrt{\rho \ln(1/\delta)}, \delta)$-differential privacy.*

Lemma 4 *If a randomized mechanism $\mathcal{M}$ satisfies (ρ, ω)-tCDP, then for any n-element dataset D, executing $\mathcal{M}$ on uniformly random sn entries ensures $(13s^2\rho, \log(1/s)/(4\rho))$-tCDP, with $\rho, s \in (0, 0.1]$, $\log(1/s) \geq 3\rho(2 + log(1/\rho))$ and $\omega \geq \log(1/s)/(2\rho)$.*

Lemma 1 connects the Gaussian mechanism to the new differential privacy definition, i.e., tCDP. It intuitively shows that by injecting the same Gaussian noise the differences between $(\epsilon, \delta)-$DP and (ρ, ω)-tCDP. Lemma 2 indicates the composition theorem of two randomized mechanisms under tCDP. Lemma 3 establishes the link between two differential privacy criteria, and Lemma 4 provides the privacy amplification via random sampling. These lemmas will serve as the basics for the proof of our adaptive random noise mechanism in the later section.

3 Evaluation of Inference Attack Models for Deep Learning on Medical Data

3.1 Introduction

Deep learning has become increasingly popular in healthcare and medicine areas. Medical institutions hold various modalities of medical data, such as electronic health records, biomedical images, and pathology test results. Based on medical data, deep neural network models are trained to address necessary healthcare concerns [1]. Examples include but not limited to: 1) deep learning models based on medical records outperformed traditional clinical models for detecting patterns in health trends and risk factors [2]; 2) deep learning model had high sensitivity and specificity for detecting diabetic retinopathy and macular edema in retinal fundus photographs [3]; 3) a mammography-based deep learning model was more accurate than traditional clinical models for predicting breast cancer risk [4].

However, the application of deep learning in healthcare is confronted with privacy threats. The medical records contain personal private information like drug usage patterns of the individual patient. Medical institutions also hold patients' profile information such as home address, gender, age, etc. The private information might be unwittingly leaked when the aforementioned data is used for training a deep learning model [7, 8]. For example, attribute inference attacks [7] can utilize the trained model and incomplete information about a data point to infer the missing information for that point. The adversary could exploit such an attack to infer the target private information according to partial information on medical records. Another instance is model inversion attacks that enable the image data used for classification inference to be recovered based on the intermediate output of convolutional neural networks (CNNs) [9]. The adversary could deploy model inversion attack to recover the medical images of any target patient and infer private health conditions accordingly.

The risk of privacy leakage makes medical institutions increasingly less willing to share their data. This inevitably slows down the research progress at the intersection of deep learning and healthcare. Thus, it is necessary to evaluate the potential hazards of various attack models on medical data and develop corresponding defenses on such attacks.

In this work, we attempt to implement two types of attack models on medical data, which are shown in Figure 1. We use attribute inference attack to infer the sensitive attributes in medical record data according to the rest attributes and class labels, when training a deep learning model. We also employ model inversion attack to recover the medical image data based on the intermediate inference output. Against those attacks, we present two types of inference attack defense mechanisms. Label perturbation provides a way to add noise into confidence scores and thus hinders the privacy leakage from model prediction. Model perturbation is proposed to add noise into parameters of deep network models and thus disturb the privacy disclosure in model inference. Experimental results show that both attacks successfully disclose the private medical information used in training and inference processes, and the attacks are not effective any more under the proposed inference attack defense mechanisms.

The main contributions of this work are summarized as follows.

- We evaluate attribute inference attack and model inversion attack on medical data. That demonstrates the privacy vulnerability of deep learning models, which limits their applications in medical area. As far as we know, we are the first to evaluate the model inversion attack on medical data.

- We present inference attack defenses based on label perturbation and model perturbation. The mechanisms can significantly alleviate the privacy breaches of medical data in both training phase and inference phase.

3.2 Related Work

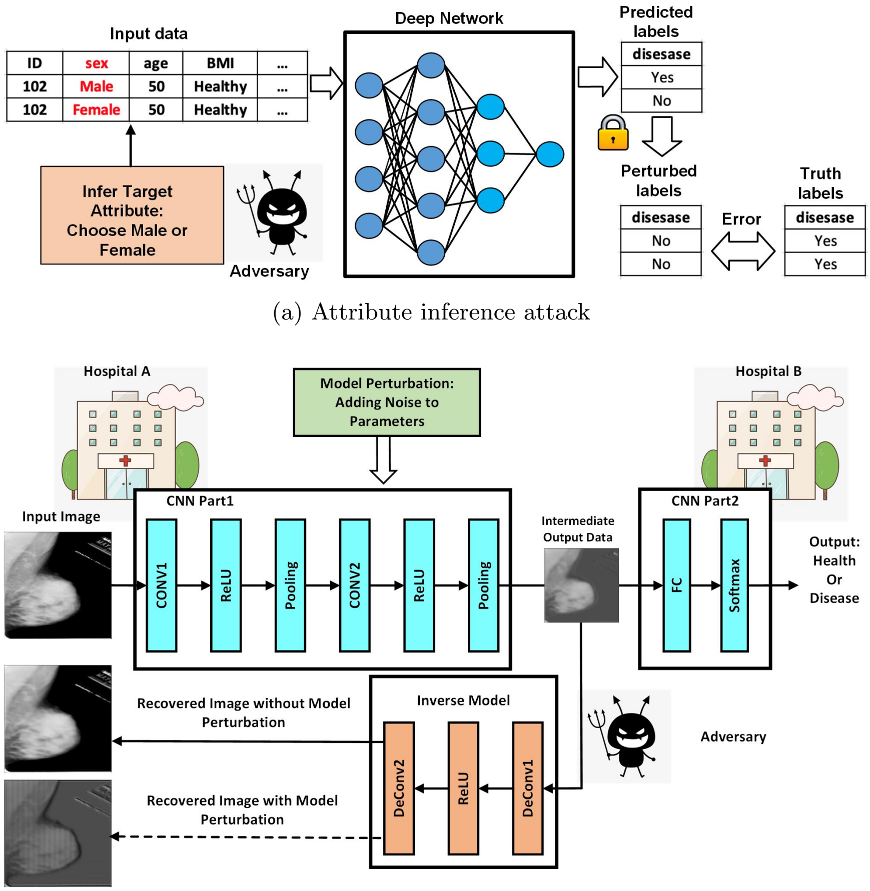
There are different types of privacy attacks against training and inference data. These attacks severely threaten patients' privacy when deep learning is used in the healthcare area. The first type is membership inference attacks [18, 8], which tries to infer whether a target sample is contained in the dataset. The second type is model encoding attacks [19], the adversary who directly accesses to the training data can encode the sensitive data into the trained model and then retrieve the encoded sensitive information. The third type is attribute inference attack, given some attributes of the dataset, the adversary could infer the sensitive attribute. The fourth type is model inversion attack, given a deep learning model and some features of input data, the adversary could recover the rest of the features of the input data. In this work, we choose two prominent inference attacks, namely attribute inference attack and model inversion attack, which may reconstruct medical images and clinical reports and be more threatening to patients' privacy. We evaluate their attack performance on medical records and medical images, and then propose defense methods against these two inference attacks.

Attribute Inference Attack Attribute inference attack is studied in various areas. Gong et al. [20, 21] studied attribute inference attacks to infer the users' sensitive attribute of social networks by integrating social friends and behavioral records. May et al. [22] proposed a new framework for inference attacks in social networks, which smartly integrates and modifies the existing state-of-the-art CNN models. Qian et al. [23] demonstrated that knowledge graphs can strengthen de-anonymization and attribute inference attacks, and thus increase the risk of privacy disclosure. However, few research evaluate the attribute inference attacks in the healthcare area. In this work, we adopt the same attribute inference attack in [7] which infers the sensitive attributes based on confident cores in predictions and is conveniently deployed on healthcare data. We propose a label perturbation method to effectually defend against the attribute inference attack.

Model inversion attack Model inversion attack is an outstanding attack to recover the input data of deep neuron networks. He et al. [7] proposed a model inversion attack to recover input images via the confidence score generated in the softmax model. He et al. [9] proposed a model inversion attack to reconstruct input images via the intermediate outputs of the neural network. Hitaj et al. [24] utilized Generative adversarial network (GAN) to recover the image in a collaborative training system. In this work, we adopt the same model inversion attack in [9] by considering the medical collaborative deep learning scenario, where two hospitals hold different parts of a deep neuron network and collaborate to complete the training and inference via transmitting the intermediate output information. As far as we know, we are the first to evaluate the model inversion attack on medical data via intermediate output information. We propose an effective and convenient perturbation method instead of using the defenses suggested in [9], i.e., combining Trust Execution Environment and Homomorphic Encryption that requires special architecture support and huge computational burden.

Other Attacks against Machine Learning Besides attribute inference attack and membership inference attack, there exist numerous other types of attacks against ML models [25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39]. A major attack type is adversarial examples [34, 32, 33, 36]. In this setting, the adversary tries to carefully craft noise and add them to the data samples aiming to mislead the target classifying. In addition, a similar type of attack is backdoor attack, where the adversary tries to embed a trigger into a trained model and to exploit when the model is deployed [27, 30, 35]. Another line of work is model stealing attack, [37] proposed the first attack on inferring a model's parameters and Other related works focus on inferring a model's hyperparameters [31, 38].

Possible Defenses To defend against the privacy attack, many researchers focused on defense methods. Trust Execution Environment [40] is specialized hardware for secure remote computation and data confidentiality protection against privileged adversaries. Homomorphic Encryption [41] allows the training and inference operations on encrypted input data, so the sensitive information will not be leaked. However, these methods require special architecture support and a huge computational burden. Differential Privacy (DP) [42] adds noise into the training model and there exits a trade-off between usability and privacy. However, our attacks mainly focus on the inference phase rather than the training phase and thus the DP methods are not suitable to defend against our attacks. We propose label perturbation that adds noise in the predicted label to defend attribute inference attack and mode perturbation that adds noise into the after-trained model to defend model inversion attack. The proposed methods are effective and convenient for application. We also give the results of the trade-off between model accuracy and attack performance. These results provide an intuitive guide for medical staff to adjust the defenses against the two inference attacks.



(a) Attribute inference attack

(b) Model inversion attack

Figure 1: Inference attack models and defense approaches for medical deep learning.

3.3 Inference Attack Models

3.3.1 Vulnerability of Medical Deep Learning

Via the use of deep learning algorithms, medical institutions can improve the rate of correct diagnosis [2]. In the training phase, deep neural networks are trained based on input medical data and output diagnosis results to learn the inherent relationships between them. In the inference phase, the after-trained deep neural networks can achieve high-accurate diagnosis results given new medical data as input. However, during the training phase and inference phase, the adversary could adopt attack methods to infer or recover the input medical data which contains sensitive information of patients. In this work, we evaluate two prominent inference attacks, i.e., attribute inference attack and model inversion attack. As for attribute inference attack, we assume that the adversary knows the attributes of all the patients except the sensitive attributes when the patients' medical records are used as input. This assumption applies to the cases that the attacker can search the rest attributes such as age and gender from the public database. The adversary utilizes the inherent relationships among attributes and labels to recover the patients' sensitive attributes of input medical records. As for model inversion attack, we take the vulnerability of collaborative deep learning as an example, which provides an efficient paradigm to accelerate the learning and prediction process. The fundamental idea is to split a deep neural network into two parts. For example as Figure 1b, in medical collaborative learning, the first few layers are stored in Hospital A while the rest are kept in Hospital B. In the collaborative training mode [43], Hospital A sends the outputs of the cut layer to Hospital B and then retrieve the gradients of the cut layer. In the collaborative inference mode [9, 44], Hospital A sends the outputs of the cut layer to Hospital B and retrieves the final results. The model training and inference processes are collaboratively carried out without sharing the raw data. However, the shared intermediate output information may be leaked during the transmission. Given the information, the adversary could recover the raw data with model inversion attack and thus compromise the data privacy of Hospital A.

3.3.2 Attribute Inference Attack

As shown in Figure 1a, attribute inference attack [7] enables an adversary to deduce sensitive attributes in patients' medical records. In this setting, the goal of the adversary is to guess the value of the sensitive features of a data point, e.g., sex attribute, given only some public knowledge about it and the model. Let (x, y) denote a data point where x denotes the input patient information, and y is the patient outcome. We assume that a deep network $f(x)$ takes the input x to predict the output y. The network's parameters are optimized by reducing the discrepancy between the predicted value $f(x)$ and the true outcome y measured by the cross-entropy loss. Let t be a sensitive variable in x that an attacker wants to learn, given the rest of the values in x. We assume that the target t is drawn from a finite set of possible values $t_1, ..., t_m$ and the adversary has the ability to obtain the marginal priors of each t_i. For each t_i, the adversary counterfactually assumes that $t = t_i$, and computes $f(x_i)$. It then computes the error between the true $f(x)$ and $f(x_i)$ with the Gaussian error function. The adversary can choose the value t_i based on the largest likelihood calculated with the Gaussian error and the marginal priors.

3.3.3 Model Inversion Attack

As shown in Figure 1b, model inversion attack [9] enables an adversary to recover an input medical image x_0 from the corresponding intermediate output $v_0 = f_\theta(x_0)$, where f_θ is the former layers of the model in Hospital A. We consider the black box attack setting, where the adversary does not know the structure or parameters θ of f_θ but he could query the black-box model, i.e., he could input the arbitrary data X into the model and observe the intermediate outputs $f_\theta(X)$. This assumption happens to the use case where Hospital A releases its APIs to other medical entities as training and inference services. In this setting, we build an inverse network model that learns the inverse mapping from output to input without the original model information. Roughly, the inverse model $g_\omega \approx f_\theta^{-1}$ can be regarded as the approximated inverse function of f_θ, where $v = f_\theta(x)$ is input and x is output.

Algorithm 1 shows the detailed model inversion attack consisting of four phases. In the observation phase, the adversary uses a cluster of samples $X = x_1, \cdots, x_n$ as inputs to query f_θ and obtain $V = f_\theta(x_1), \cdots, f_\theta(x_n)$. Here the sample set X is assumed to follow the same distribution of x_0. The assumption applies to the case that the radiologic images usually follows the same distribution. In the training phase, the adversary trains the inverse network g_ω by using V as inputs and X as targets. We exploit the l_2 norm in the pixel space as the loss function, which is given as

$$l(\omega; X) = \frac{1}{n}\sum_{i=1}^{n} \left\| g_{\theta 2}\big(f_{\theta 1}(x_i)\big) - x_i \right\|_2^2 . \tag{4}$$

In particular, the structure of g_ω is not necessarily related to f_θ. In our experiment, an entirely different architecture is leveraged for the attack. In the recovery phase, the adversary leverages the trained inverse model to recover the raw data from the intermediate value: $x'_0 = g_\omega(v_0)$.

Algorithm 1 Model Inversion Attack Algorithm

```
Input: input data $X = x_1, x_2, \cdots, x_n$ of the same distribution from target data $x_0$, output $v_0$ of target data, batch size $B$, epoch number $E$, learning rate $\eta$
Output: recovered data $x'_0$
query the model by input data $V = f_\theta(X)$
initialize $\omega_0$
for each epoch $t \in T$ do
    $\beta \leftarrow$ (split $V$ into batches of size $B$)
    for each batch $b \in \beta$ do
        $\omega_{t+1} \leftarrow \omega_t - \eta \nabla l(\omega_t; b)$
    end for
end for
recover the target data $x'_0 = g_\theta(v_0)$
return $x'_0$
```

3.4 Inference Attack Defense Mechanisms

3.4.1 Label Perturbation Based Protection

We apply randomized responses [45] to protect the learning model output labels of each data sample. Intuitively, given a flipping probability p, for a binary classification, the predicted label y will be flipped with p. Similarly, if there are C $(c > 2)$ classes, there is p probability that the predicted label y_i will be replaced. If the predicted label y_i is going to be replaced, there is $1/(c-1)$ probability that the label y_i will be substituted by the label y_j for all $j \neq i$. The inference accuracy of the attribute inference attack will deteriorate when the adversary obtains the inaccurate predicted label. Although the training performance can be influenced by the label perturbation, by controlling the flipping probability p carefully, we can still have an acceptable training model.

3.4.2 Model Perturbation Based Protection

To defend against model inversion attack, we adopt model perturbation in CNN model. Different from the label perturbation that adds noise into predicted label, model perturbation adds noise into model parameters θ (weights and bias) before the forward propagation is implemented. Specifically, we use Gaussian mechanism with expectation 0 and variance σ to generate noise and add it into model parameters, which is given as

$$\theta = \theta + \mathcal{N}(0, \sigma^2 I). \tag{5}$$

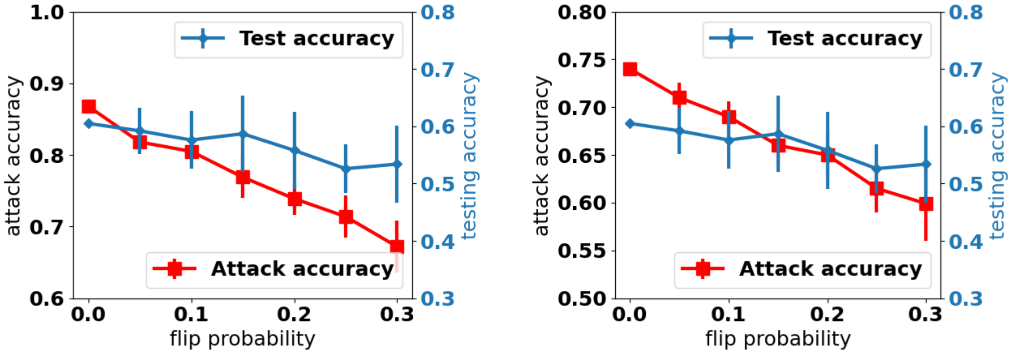
Accordingly, the output of the cut layer is perturbed in collaborative deep learning. Model inversion attack becomes difficult to build an accurate mapping from the output to the input image and thus the image recovery quality decreases.

3.5 Performance Evaluation

3.5.1 Attribute Inference Attack

Experiment Settings We evaluate attribute inference attack and label perturbation approach on two public medical record datasets: cardiovascular disease dataset and heart disease dataset. The cardiovascular disease dataset consists of $70,000$ records, 11 feature attributes including sensitive information such as age and gender, and labels indicating the presence or absence of cardiovascular disease. Heart disease dataset [46] contains 13 attributes, 303 instances, and labels referring to the presence of heart disease in individual patients. We split the dataset into training set and testing set with 80% and 20%. Our experiments use a neural network with 4 fully connected layers.

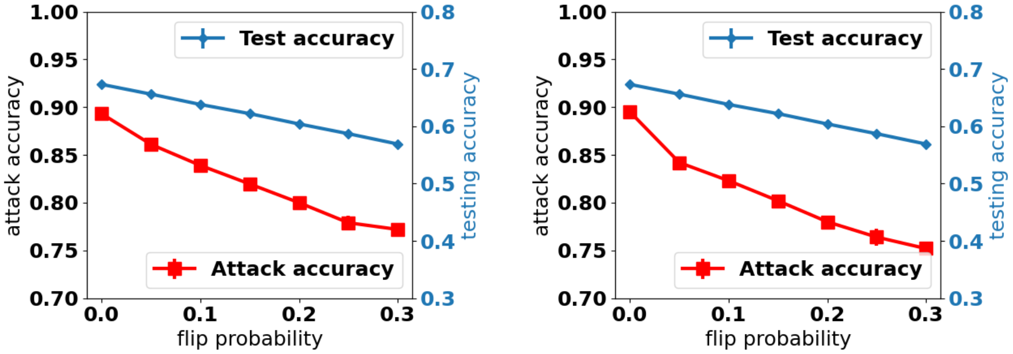
Evaluation Results As described in Section 3.3.2, in the experiments, we assume the attacker can obtain other information of a patient except for only one attribute and the marginal prior knowledge of the targeted attribute. We implement each experiment 10



(a) Fasting blood sugar

(b) Gender

Figure 2: Attribute attack performance on the "heart disease" dataset.



(a) Smoking

(b) Alcohol intake

Figure 3: Attribute attack performance on the "cardiovascular disease" dataset.

times and show the mean value as the curve and the standard deviation as the error bar. The flip probability denotes the defense level. The higher flip probability means better defense. When the flip probability is equal to 0, it means no defense mechanism is applied. Figure 2 and 3 demonstrates the attack and defense performance on two datasets. We select two attributes from the "heart" dataset, fasting blood sugar and gender, as the attacker's targets. For the "cardiovascular" dataset, we choose smoking and alcohol intake as the target attribute. We can observe that the attack accuracy reduces with higher flip probability. Also, the testing accuracy degrades slightly, if the defense level is high.

3.5.2 Model Inversion Attack

Experiment Settings We evaluate model inversion attack and model perturbation-based defence on two public mammography datasets: MIAS [47] and CBIS-DDSM [48]. All the images of MIAS dataset have been padded/clipped to 1024 × 1024. A total of 280 samples are obtained from MIAS for training (181 normal, 57 benign, 42 malignant) while 50 samples are used for testing (26 normal, 12 benign, 12 malignant). We clip and compress all the images of CBIS-DDSM to 256 × 256. A total of 2, 326 samples are obtained from CBIS-DDSM for training (1, 263 benign, 1, 063 malignant) while 772 samples are used for testing (419 benign, 353 malignant). We adopt an CNN with 6 convolution layers and 2 fully connection layers on the two datasets. Each convolution layer has 32 channels and kernel size is 3. There is a maxpool layer after every two convolution layers. The model is split at 2nd, 4th, and 6th convolution layers. We select ADAM as our optimizer and set the learning rate as 0.001.

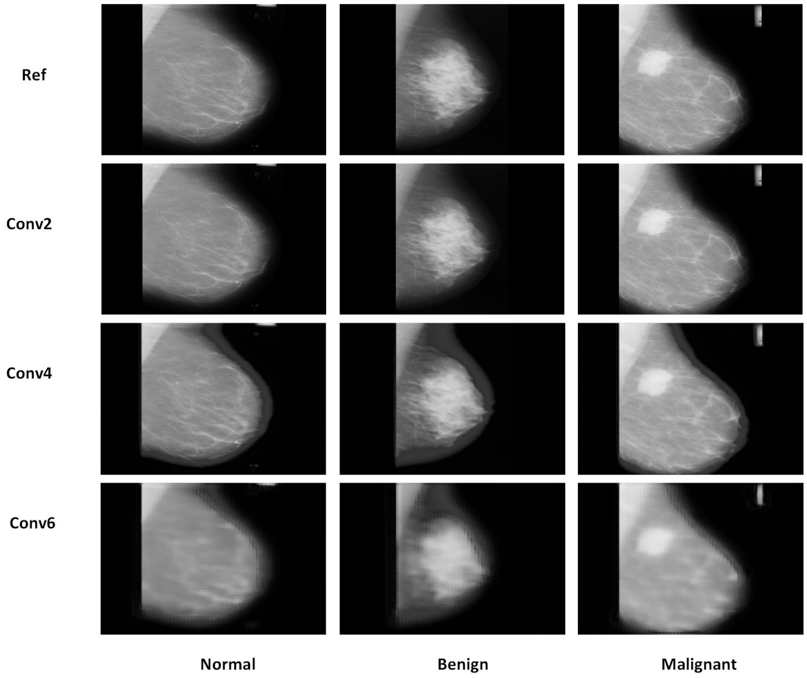


Figure 4: Recovered MIA inputs via model inversion attack.

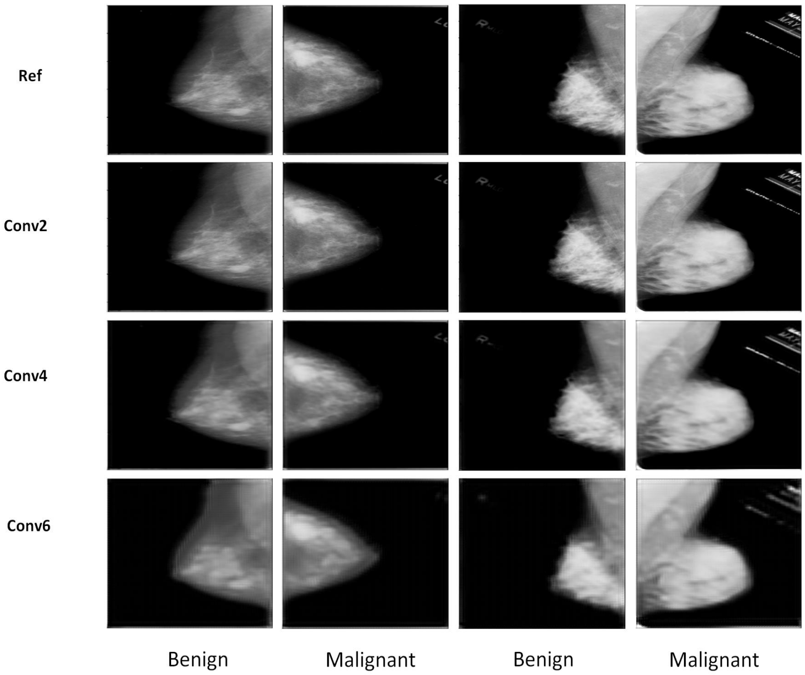


Figure 5: Recovered CBIS-DDSM inputs via model inversion attack.

Table 1: MSE, PSNR, SSIM for model inversion attack with different split layers

	MIAS			DDSM		
	layer 2	layer 4	layer 6	layer 2	layer 4	layer 6
MSE	2.925	55.042	88.839	1.672	29.649	110.460
PSNR	44.162	31.039	28.995	46.385	33.962	28.269
SSIM	0.999	0.994	0.990	0.999	0.995	0.984

Evaluation Results Figure 4 and Figure 5 show the recovered images via model inversion attack. When the split point is in a lower layer, the recovered images have high quality. When the split point is in a deeper layer, the recovered images become relatively blurry and lose certain details. But even if in the recovered image from the output of the 6th layer, the details within the breasts can still be clearly identified. The attackers could diagnose the patient's breast health given the recovered mammography image with the help of classification models or radiologists.

To quantify the attack results, we adopt three metrics, Mean-Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [49], which are shown in Table 1. MSE reflects pixel-wise similarity while PSNR measures the pixel-level recovery quality of the image. SSIM measures the human perceptual similarity of two images by considering their luminance, contrast, and structure. It ranges from $[0, 1]$, where 1 represents the most similar. When the split point is in a deeper layer, the recovered inputs have higher MSE, PSNR, and lower SSIM, which means the attack becomes harder.

Figure 6 and Figure 7 show the defense performance of model perturbation with different noise scale when the split point is in the 4th layer. We experiment Gaussian noise distributions with scale 0.02 to 0.05 and central 0. When the scale increases, the recovered inputs become more blurry and lose more details.

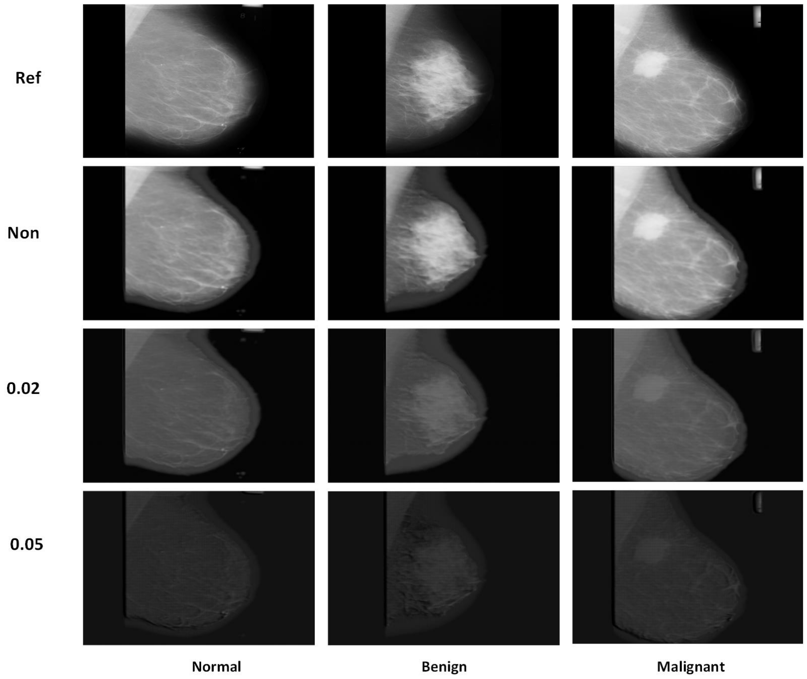


Figure 6: Recovered MIAS inputs with and without model perturbation.

3.6 Conclusion

In this work, we have evaluated two types of inference attacks on medical images and clinical records, and demonstrated that these attacks can infer sensitive attributes of medical health records as well as recover medical images at high fidelity. Our research finding exposes the risk of privacy leakage for using deep learning models in training medical data. To circumvent this problem, we proposed inference attack defenses based on label perturbation and model perturbation. Experimental results showed that the proposed defenses can

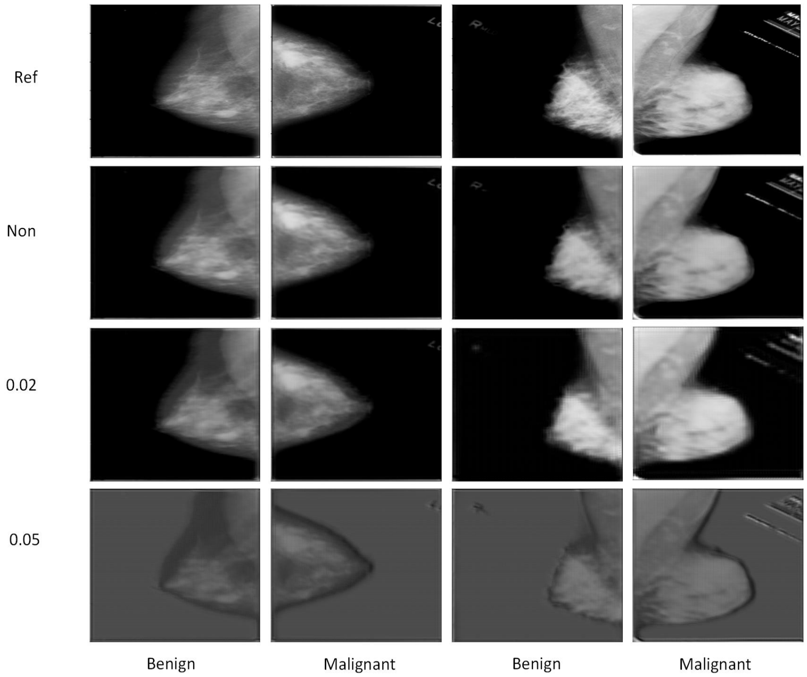


Figure 7: Recovered CBIS-DDSM inputs with and without model perturbation.

effectively defend the malicious inference attacks while the deep learning performance can still be preserved commendably. The experimental results and the approaches presented help to raise awareness about the privacy issues of deploying deep learning networks in medicine and potentially open up a new vista to ensure patients' privacy and confidentiality in the increasing adaptation of AI-enabled information infrastructure in healthcare delivery and medical research.

4 Adaptive Privacy Preserving Deep Learning Algorithms for Medical Data

4.1 Introduction

Deep learning holds great promise in improving healthcare and medicine. Examples include but not limited to: i) deep neural networks have exceeded expert performance on referral recommendation of sight-threatening retinal diseases [50]; ii) convolutional neural networks trained with more than 100,000 radiographs have shown competitive diagnostic accuracy compared to six board-certified radiologists while being two orders of magnitude faster [51]. Accenture estimates that artificial intelligence, in which deep learning is a crucial component, could save the healthcare industry $150 billion annually by 2026. For deep neural networks to work well, they need to be trained with a large number of examples. Unfortunately, sensitive information, including patient images and electronic health records, can be reconstructed with high fidelity from deep neural networks using privacy attacks during or after the network training process. To make the situation worse, the common strategy of data anonymization is not safe enough because adversarial parties can re-identify individuals in anonymized datasets by combining the data with background information. A notable experiment shows that combining public anonymized medical records and voter registration records can successfully identify the personal health information of a former Massachusetts governor, which is called linkage attack [52].

There are several popular types of attacks, such as (i) attribute attacks [53] which infer sensitive pieces of information (e.g, whether a patient has cancer) given the patient's public record and the ability to query the machine learning model; (ii) membership inference attacks [8] whose goal is to find out if a patient record is in the pool of the data used to train the machine learning model; and (iii) model inversion attacks [7] which attempt to reconstruct the entire patient data given only access to an intermediate layer of the deep network. As the medical records contain patients' sensitive data, realizing the full potential of deep learning in healthcare requires an innovative approach for building and deploying deep neural networks without sacrificing patients' privacy.

Differential privacy (DP) [14] as a golden standard of privacy provides strong guarantees on the risk of compromising the sensitive users' data in machine learning applications. Intuitively, it works by adding random noise to the model parameters so that an adversary with arbitrary background knowledge cannot confidently conclude whether a users' data is used in training a model or not. There are many papers focusing on designs for differentially private machine learning algorithms including empirical risk minimization and deep neural networks. The approaches to achieve private empirical risk minimization mainly include: output perturbation [54, 55, 56, 57] (add DP noise to model parameter obtained after the training), objective perturbation (add DP noise to objective function) [54, 58, 59], and gradient perturbation [60, 61, 62, 63] (add DP noise to the gradient). Note that the output and objective perturbation methods require the (strong) convexity of the objective function, which makes them impossible to apply in deep learning problems. Hence, injecting differentially private noise into gradient is a proper way to obtain a private deep learning model. The first work employed gradient perturbation method to achieve differential privacy on deep learning is called differentially private stochastic gradient descent (DPSGD) algorithm [64], which is also adopted by Google's TensorFlow Privacy. Since the gradient norm is usually unbounded in deep network optimization, gradient perturbation can be used after manually clipping the gradients at each iteration. In [64], the authors utilized norm gradient clipping to bound the effects of an individual data sample on the gradients, which is required for generating noise in the gradient perturbation method. Then, the differentially private noise is injected into the clipped gradient. As we can update the gradient of each step differentially privately, it is guaranteed that the overall deep learning model is private. Although [64] utilizes the moments accountant method to achieve a tight analysis of the privacy loss over the large number of iterations, the classification performance of DPSGD is still far inferior to the original SGD.

In this work, we aim to build an accurate deep learning model without compromising medical data privacy. To be specific, we first clip the gradient with l_2 norm and then inject linear decaying Gaussian noise to the gradient of each step. Our salient contributions are summarized as follows.

- We propose a novel adaptive differentially private deep learning algorithm to protect medical training data. Intuitively, compared with the DPSGD algorithm, the advantages of the proposed algorithm include: a) We carefully adjust the scale of noise in each iteration controlled by a decay rate to reduce the negative noise addition and guarantee the convergence property of deep learning algorithm; b) instead of using the moments accountant applied in DPSGD [64], we adopt the truncated concentrated differential privacy (tCDP), which provides a simple, explicit, and tight privacy bound analysis on adaptive noise injection while avoiding the numerical computation of log moments. Moreover, tCDP can provide privacy amplification via random sampling compared with zero concentrated differential Privacy (zCDP) [65].

- We evaluate the performance of the proposed adaptive DP deep learning algorithm based on real-world chest radiographs. As far as we know, this is the first work focusing on multi-label classification tasks on medical datasets. We compare the performance of the proposed model with DPSGD on the same privacy preserving level. Our extensive experimental results show that the convergence of the proposed model is faster and the accuracy is higher. Moreover, our hyperparameter settings may pave the way for further the application of differentially private deep learning in medical domains.

4.2 Related Work

4.2.1 Privacy threats in machine learning

Many attack models have been proposed in the literature. The membership inference attack [8] is proposed to infer whether the training dataset consists of a specific data sample. Fredrikson et al. introduced model inversion attack in [7], where the adversary can reconstruct training samples with some known features and the access to the machine learning model. In [66], the authors proposed a power side-channel attack model to recover the input data. Tramer et al. proposed the model stealing attack [37], where the adversary only has the access to a target model but not has any other knowledge of the model, and aims to generate a model that has similar performance of a target model. Moreover, other works focus on inferring the hyperparameters of the learning model [31, 38].

4.2.2 Privacy preserving empirical risk minimization

Recently, many researchers focus on private empirical risk minimization (ERM) problems [67, 68]. In [69], the authors designed a differentially private algorithm for online linear optimization problems with optimal regret bounds. The authors in [70] investigated the relationships between learnability and stability and privacy and concluded that a problem is privately learnable only when existing a private algorithm that can asymptotically minimize the empirical risk. [71] proposed private incremental regression and a private incremental ERM problem combining continual release to analyze the utility bound of several algorithms. In [72], the authors provided small excess risk in the generalized linear model with sampling based method for entropy regularized ERM. There are also some papers targeting at private ERM learning on high dimensional datasets. The authors in [73] provided differentially private algorithms for sparse regression problems in high-dimensional settings. Smith et al. [74] used an algorithm based on a sample efficiency test of stability to extend and improve the results. In [75], the authors introduced Gaussian width of the parameter space in the random projection to derive a risk bound by using a private compress learning method in ERM algorithms. In distributed machine learning, [62, 76, 77] proposed differentially private alternating direction method of multipliers (ADMM) algorithms with Gaussian mechanism.

4.2.3 Privacy preserving deep learning

As differential privacy can provide strong privacy guarantee, differentially private deep learning models have attracted enormous attentions. Abadi et al. [64] proposed the differentially private stochastic gradient descent (DPSGD) algorithm and adopted moments accountant (MA) to calculate the overall privacy budget. However, there is no closed-form mathematical expression to estimate privacy budget. In order to improve the utility of the DPSGD while preserving privacy, the authors in [65, 78] designed several adaptive differentially private deep learning models by allocating different privacy budgets to each iteration and employed zero-concentrated differential privacy (zCDP) to analyse the privacy loss during the training. The difference between this paper and our proposed model is the design of the decay function and the DP definition. In our work, we adopted tCDP for privacy bound analysis. The tCDP is the relaxation of zCDP, which can provide privacy amplification via random sampling compared with zCDP. In [79], the authors trained an ensemble teacher model by combining a set of teacher models, which are trained over disjoint training datasets and the author also trained the differentially private student model by querying the ensemble teacher to label public data. Furthermore, Xie et al. [80] and Zhang et al. [81] focused on achieving differential privacy on Generative Adversarial Nets (GAN). In [57, 81], the authors injected differentially private noises to the loss function based on the functional mechanism. However, none of these works provide utility guarantees for their algorithms.

4.3 Adaptive Differentially Private Deep Networks

4.3.1 Threat model

Before presenting the adaptive differentially private deep learning model, we first describe the threat model. As DP can provide strong privacy guarantee, it is a worst-case notion of privacy. DP ensures that although attackers can have all information from the training dataset except one data sample, they still cannot get this data sample after launching attacks [14]. Specifically, in this work, we consider the white-box attack [9] where the adversary has the full knowledge of our deep networks, including their architectures and parameters. In other words, attackers can access to the published model instead of the training process. The goal of the proposed scheme is that even though the attackers have the ability to obtain other data samples in the training dataset, they cannot infer the target training data sample.

4.3.2 Privacy preserving deep learning model

We assume there are m training data samples and each data sample is denoted by $\{x_i, y_i\}$, where y_i is the label. The loss function of the training model with parameter w is defined as $L(w, x)$. The gradient of the loss function $\nabla L(w, x)$ is updated by stochastic gradient descent (SGD) during each iteration. In order to preserve privacy of the training data, the differentially private noise is supposed to add to the gradient in each iteration. Based on Theorem 1, when calculating how much noise needs to be injected into the gradient, it is supposed to have the sensitivity of the gradient, which is difficult to characterize. Therefore, we control the sensitivity by clipping the gradient in l_2 norm. With a clipping threshold C, we can replace the gradient g_t of each step by $\frac{1}{s}\sum_{i\in S_t}\Big(\nabla L(w_t, x_i, y_i)/\max(1, \frac{||\nabla L(w_t)||_2}{C})\Big)$, where s is the batch size. Then, we can add Gaussian noise to the clipped gradient. Consequently, each SGD step is considered as differentially private. Based on the composition theorem of differential privacy, the overall model is supposed to be differentially private with accumulated privacy budget.

When injecting Gaussian noise to the gradient, the privacy budget will be accumulated due to the iterations within each epoch as described in Lemma 2. If the total privacy budget is certain, we need to allocate it to each training step. The noise scale of Gaussian mechanism is decided by the privacy budget allocated to each epoch, which influences the final model accuracy. Our purpose is to achieve better accuracy of the differentially private training model without compromising data privacy. Therefore, we propose the adaptive differentially private deep learning model, which is inspired by the adaptive learning rate strategy. During the practical training processes, the learning rate is recommended to be decreased instead of fixed, in order to improve the model performance. Hence, in the DP learning model, we propose to reduce the injected noise along with the training iterations. In other words, in order to increase the accuracy, it is intended to add smaller and smaller noise to the gradients through the training time. Therefore, we propose the adaptive differentially private deep networks by injecting linear decaying Gaussian noise to the gradient during the training.

The overall procedure of our mechanism is shown in Algorithm 2. Note that we adopt tCDP in our algorithm instead of approximately differential privacy since its composition property is more straightforward for our adaptive noise addition. In each iteration of our algorithm, a batch of examples S_t with size s is sampled from the training dataset, and the algorithm computes the gradient of the loss on the examples in the batch and uses the average in the gradient descent step. The gradient clipping bounds per-example gradients by l_2 norm clipping with a threshold C. After gradient clipping, the sensitivity of the average gradient is $\frac{2C}{s}$. We next inject *linear decaying Gaussian noise* to the gradients at every training iteration with a decay rate R. This is in contrast to the mainstream approach adopted by Google's TensorFlow Privacy, which employs the same noise scale in each step of the whole training process. Specifically, we apply the Gaussian mechanism to add random noise following $\mathcal{N}(0, \sigma_t^2 \mathbb{I})$ distribution to the network's gradients. The noise variance varies with a linear decay model as $\sigma_{t+1}^2 = R\sigma_t^2$, where $R \in (0, 1)$. Moreover, by considering the privacy composition between iterations and privacy amplification by sampling, the privacy guarantee of Algorithm 2 is provided in the next section.

4.3.3 Privacy guarantee

We employ the composition theorem of Truncated Concentrated Differential Privacy (tCDP) to analyze the cumulative privacy loss of differentially private stochastic gradient descent (DPSGD), which was developed to accommodate a larger number of computations and provides a sharper and tighter analysis of privacy loss than the strong composition theorem of (ϵ, δ)-DP. One popular way to track the privacy loss of DPSGD is the Moments Accountant (MA) method [64], which is adopted by Google's TensorFlow Privacy. As for the proposed approach, a Gaussian mechanism with a linearly decaying variance is applied to DPSGD to improve the model accuracy.

Theorem 2 *Algorithm 2 provides (ϵ, δ)-differential privacy.*

Proof. Since the utilization of Gaussian Mechanism, each iteration is $\rho_t = 2C^2/(s^2\sigma_t^2)$-tCDP (Lemma 1). By Lemma 2 and Lemma 4[1], and the decay rate R of noise scale, we derive that the total privacy loss is $(\rho_{total}, \omega_{total})$-tCDP with

$$\rho_{total} = \frac{13(s/m)^2 C^2 (1 - R^T)}{2\sigma_0^2 (R^{T-1} - R^T)} \tag{6}$$

and

$$\omega_{total} = \frac{\log(m/s)\sigma_0^2 R^T}{2C^2}, \tag{7}$$

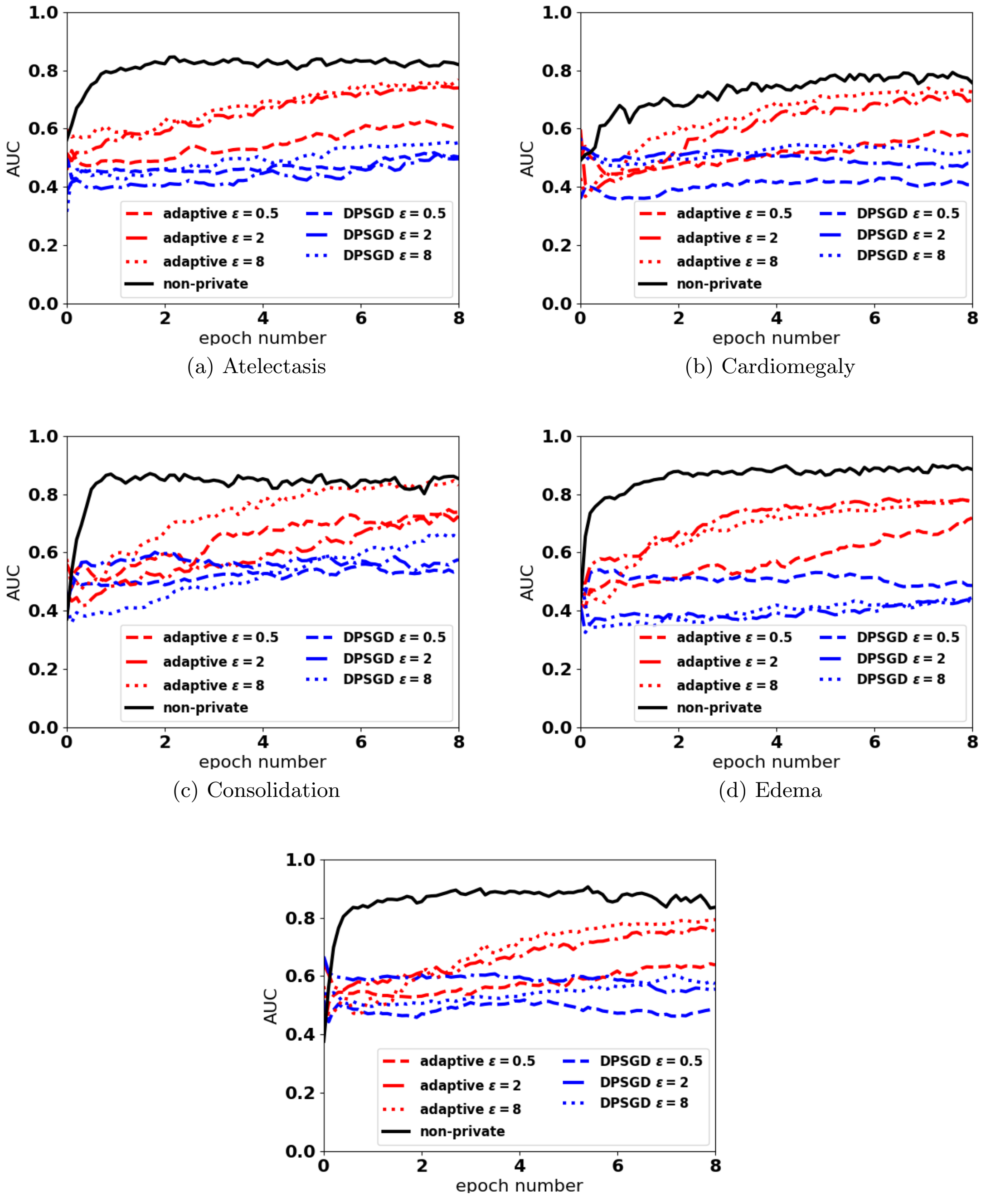
where s is the batch size, m is the total number of private training dataset, C is the clipping threshold. By utilizing Lemma 3, we can say that our algorithm satisfies $(\rho_{total} + 2\sqrt{\rho_{total}\log(1/\delta)}, \delta)$-DP, which means $\epsilon = \rho_{total} + 2\sqrt{\rho_{total}\log(1/\delta)}$. □

Algorithm 2 Adaptive Differentially Private Deep Learning

```
Input: Private training dataset $\{x_i, y_i\}_{i=1}^m$, loss function $L$, learning rate $\eta$, gradient norm bound $C$, decay rate $R \in (0, 1)$, batch size $s$.
Output: Differentially private model $w_T$.
Initialize $w_0$, $\sigma_0^2$.
$t = 0$.
for $t = 0, \cdots, T - 1$ do
    Randomly take a batch of data samples $S_t$ from the training dataset with $|S_t| = s$.
    Compute the gradient with gradient clipping
    $g_t = \frac{1}{s}\sum_{i \in S_t}\Big(\nabla L(w_t, x_i, y_i)/\max(1, \frac{||\nabla L(w_t)||_2}{C})\Big)$.
    Add adaptive Gaussian noise $g_t = g_t + \mathcal{N}(0, \sigma_t^2 \mathbb{I})$ with $\sigma_{t+1}^2 = R\sigma_t^2$.
    Update the model parameter $w_{t+1} = w_t - \eta g_t$.
end for
```

Compared with MA, our proposed approach provides an explicit closed-form mathematical expression to approximately estimate the privacy loss. It is easy to compute and can be useful when the users would like to decide proper training time, noise scale, and sampling ratio during the planning phase.

[1] Several conditions for privacy amplification via sampling (Lemma 4) are required.



(a) Atelectasis

(b) Cardiomegaly

(c) Consolidation

(d) Edema

(e) Pleural effusion

Figure 8: Comparison between non-private model, DPSGD and our adaptive DP model.

4.4 Performance Evaluation

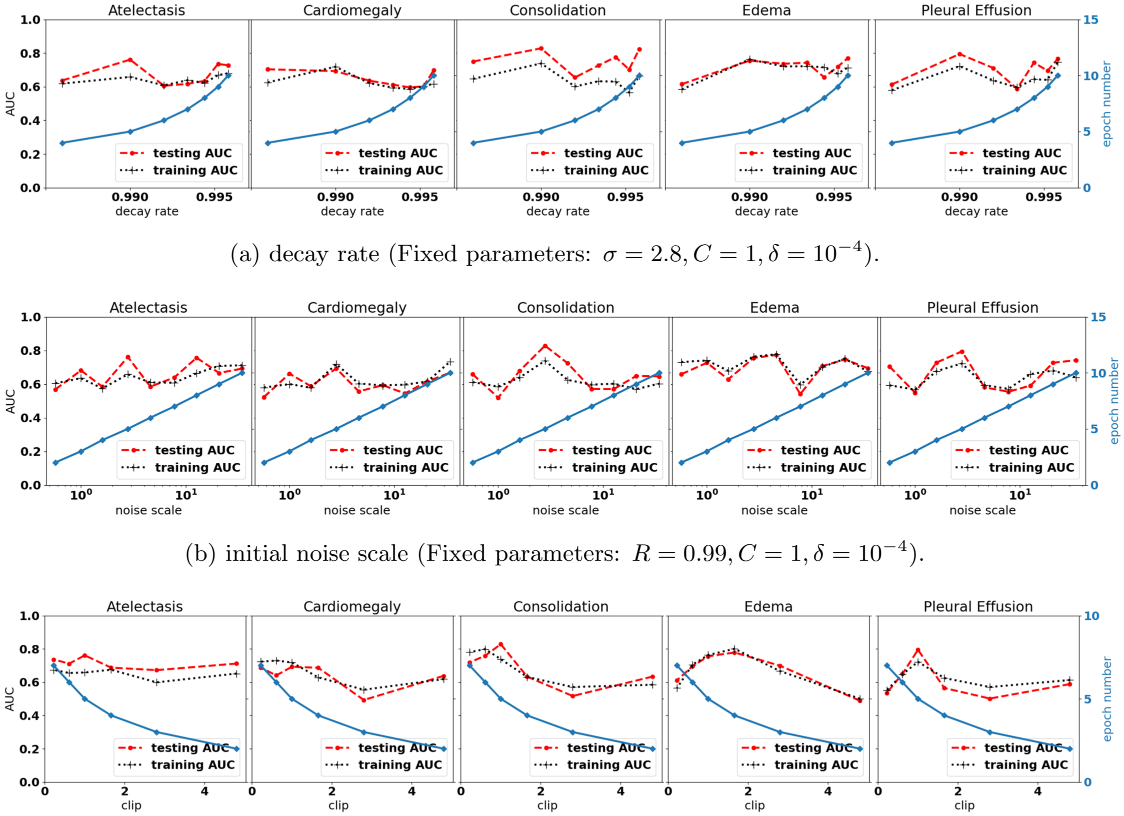
In this section, we demonstrate the experimental results of our proposed scheme on one medical dataset CheXpert, and two popular image datasets MNIST, and CIFAR10.

4.4.1 Experiment settings

CheXpert. We conduct experiments on the CheXpert dataset [82], which is a large dataset containing 224,316 chest X-rays of 65,240 patients. There are 5 classes corresponding to different thoracic pathologies: (a) Atelectasis, (b) Cardiomegaly, (c) Consolidation, (d) Edema, and (e) Pleural Effusion. The images with size 320×320 pixels are fed into the pre-trained DenseNet-121. We only re-initialize the fully connected layer and fix the other convolutional layers, which will not have influences on the privacy leakage [64]. For illustrative purposes, we use 10,000 radiographs for training and 234 for testing. We use the test set provided by Stanford CheXpert to make our results comparable to those in the literature. The test set is small because each sample requires manual annotations by 3 board-certified radiologists to create the ground truth label. For this reason, we cannot mix up noisy labels from the training set with well-validated labels from the test set [82].

Here, we introduce the default values of different parameters in the proposed adaptive differentially private deep learning model. We set the batch size s as 100, and the sample rate $\frac{s}{m} = 0.01$. We assume that the gradient norm clipping threshold C is 1, the initial noise scale σ_0 is 2.8, the noise decay rate R is 0.99, and the broken probability δ is 10^{-4}. Recall the analysis in Section 4.3.3, we can obtain the relationship between these parameters and the privacy with equations (6) and (7). As long as we have fixed the privacy budget ϵ, we can easily calculate the other parameters with these equations. We employ the area under the curve (AUC) to evaluate the non-private and private deep learning models.

MNIST The handwritten digits dataset [83] consists of 60000 training images and 10000 testing images, which are 28×28 gray scale images. We stack two convolutional layers with max-pooling and two fully-connected layers. Instead of using ReLU as the activation functions, we use tanh in the MNIST model as suggested in [84], which can provide better



(a) decay rate (Fixed parameters: $\sigma = 2.8, C = 1, \delta = 10^{-4}$).

(b) initial noise scale (Fixed parameters: $R = 0.99, C = 1, \delta = 10^{-4}$).

(c) clip threshold (Fixed parameters: $R = 0.99, \sigma = 2.8, \delta = 10^{-4}$).

Figure 9: The impact of parameters on the training model performance (Red curve is tesing AUC, black curve is training AUC and the blue curve is epoch number).

performance.

CIFAR10 The CIFAR10 [85] dataset consists of 10 classes 32×32 color images. There are 50000 training examples and 10000 testing examples. We use the pretrained ResNet18 as the training model for this dataset and re-initialize the fully connected layer. Then, we train the model with the proposed mechanism.

4.4.2 Experiment results

CheXpert. As introduced in section 4.4.1, there are 5 labels of each data sample in the dataset. Hence, we show five figures for each experiment. Firstly, we compare the testing AUC of the proposed adaptive model with DPSGD and non-private model in Figure 8. In the experiment, we set the epoch number as 8. The privacy budget ϵ varies according to different initial noise scales in the DPSGD and the proposed adaptive DP model. The figure shows that with a higher epsilon value, the model accuracy is lower, since a larger epsilon means less noise injected to the gradient. We can also observe that our proposed adaptive DP model outperforms the DPSGD model for all thoracic conditions across different privacy budgets. More specifically, with the same privacy budget $\epsilon = 2$ and 8, the adaptive DP model can reach the average of 80% testing AUC for all of the five labels, while the DPSGD only can achieve approximately 60% testing AUC. We can conclude that with adaptive DP model, the performance will not drop too much and the patient's privacy is preserved.

In Figure 9, we demonstrate the impact of different parameter settings of the proposed adaptive differentially private deep learning model. We explore the influences of four parameters, noise decay rate R, initial noise scale σ_0, and clipping threshold C on the performance of the adaptive DP model. In the experiments, we keep the privacy budget fixed as $\epsilon = 8$. When the experiment is focusing on a specific parameter, we only vary values of this parameter and adjust the number of epochs to maintain the fixed privacy budget. In other words, only the discussed parameter and epoch number change in each experiment. With a larger noise decay rate, a higher initial noise scale, or a lower clipping threshold, it costs less privacy budget during each iteration. Therefore, we can achieve more epochs during training as shown as the blue solid curve in Figure 9. Moreover, we can observe that the model performance is better with a higher decay rate, a higher initial noise scale, and a smaller clip threshold, since the privacy budget spends slower as the number of training epochs increases.

MNIST and CIFAR10. We repeat the experiments on MNIST and CIFAR10 datasets and the experimental results are shown in Table 2. We compare the test accuracy by applying DPSGD and the proposed adaptive model. We first train the DPSGD model under a desired epoch number, keep the privacy budget ϵ value and calculate the parameters of the adaptive model with equations (6) and (7). For MNIST dataset, the test accuracy of non-private model can reach 99%. With the privacy budget ϵ equal to 1.19, 3.01, and 7.1, test accuracy of the proposed adaptive model is 1.09%, 0.25% and 0.2% higher than that of

Table 2: Summary results on MNIST and CIFAR10.

Dataset	Privacy Budget (ϵ, δ)	Accuracy	
		DPSGD	Proposed
MNIST	Non-Private	99%	
	(1.19, 10^{-5})	96.61%	97.7%
	(3.01, 10^{-5})	97.82%	98.07%
	(7.1, 10^{-5})	97.97%	98.17%
CIFAR10	Non-Private	88.67%	
	(3.02, 10^{-5})	77.16%	83.15%
	(7.03, 10^{-5})	81.42%	84.3%

DPSGD. For CIFAR10 dataset, the non-private model can get to 88.67% test accuracy in 90 epochs. Compared with the DPSGD model, with the privacy budget ϵ of 3.02 and 7.03, test accuracy of the proposed adaptive model is promoted by 5.99% and 2.88%.

4.5 Conclusion

In this work, we propose the adaptive differentially private deep learning model. Intuitively, we first clip the gradient to bound the sensitivity, inject differentially private noise with a specific decay rate based on the Gaussian mechanism into the clipped gradient, and update the gradient with SGD. The proposed algorithm is easy to implement and significantly improve the performances on various well-known datasets. Because of the large number of iterations in deep learning model, we adopt tCDP to obtain a tight bound of privacy leakage, since tCDP can provide a tighter and closed-form mathematical expression to estimate privacy budget compared with MA. Furthermore, tCDP can provide privacy amplification via random sampling compared with zCDP. We also conduct experiments on the public CheXpert dataset to verify the effectiveness of our adaptive differentially private deep learning model. We aim to explore the potential of adaptive differentially private deep learning applications in medicine. Moreover, we used the CheXpert that is a multi-label classification task. As far as we know, there are no works focusing on medical datasets with multi-label classification tasks.

5 Differentially Private Functional Mechanism for Generative Adversarial Networks

5.1 Introduction

With the rapid development in cyber-physical systems and IoT systems, there is more and more data available for data analysis tasks providing opportunities for new innovations. In order to glean useful insights from huge data sets, many statistical models such as data mining, machine learning etc., have been developed. With the statistical models being developed and analyzed, it is certain that training data with large size are yielding desirable results. Although huge amount of data is available and accessible in the big data era, there are still some domains that have limited data for training, thereby affecting the performance of the models. For instance, in the medical field, there is lack of data availability to train a promising deep learning model on a particular disease, as it is impossible to access all patients' medical records to get the training data samples while obeying the privacy terms and conditions.

Generative adversarial network (GAN), first proposed in [5] is a potential solution that addresses the data scarcity problem in fields with limited data because of its ability to generate data similar to real data. Typically, GAN integrates deep learning and game theory to train two models, generator and discriminator, so that the generator can generate high fidelity fake samples from latent space, that are supposed to be indistinguishable from the real data samples. Despite the solid fundamental theories and impressive empirical experiments, training the original GAN is relatively tricky and improving the quality of generated images is still challenging. Deep convolutional GAN (DCGAN) proposed in [6] first attempted to deal with this problem by applying convolutional neural networks (CNNs) in GAN to enable stable training and improve the applicability of the generated images. GANs and its variants are being investigated to achieve effective results through data augmentation and overcome training data insufficiency for the models.

Enormous synthetic data generated closer to real data by GANs, along with the deep learning algorithms is helpful in extracting meaningful information from data sets. GANs can easily remember the training samples due to the high model complexity of deep neural networks, since the density of learned generative distribution concentrates on the training data points. Therefore, GANs without privacy considerations can have undesired drawbacks and critical concerns, especially when training a model with sensitive data such as patients' health records. For example, Arjovsky et al. in [86] presented that the training samples could be recovered by repeatedly sampling from the distribution, and through active inference attacks adversaries can learn the training samples from the generated high quality data [24]. Thus, the data privacy problem in machine learning, especially the privacy of the training datasets, is attracting more research attention since personal data is a valuable asset in the data era.

To protect personal data, privacy preserving data publishing techniques should be implemented, and models should be trained on such datasets with utility maximization as the objective. Differential privacy is a state-of-the-art privacy preservation mechanism, which can measure the difference in the output of an algorithm due to the presence/absence of a single element in the original dataset. Therefore, it can ensure that the adversary cannot infer any sensitive information. Papernot et al. in [87] employed teacher student model in deep learning to provide privacy guarantee for training data. Nevertheless, the privacy loss is proportional to the number of data in the public data set, which needs to be labelled. Xie et al. in [80] proposed a differentially private GAN (DPGAN), where the Gaussian noise is injected into the gradient of discriminator during the training procedure. This method is proved to be powerful in [88]. However, the privacy budget accumulates with the training steps, which implies that the data privacy will be disclosed due to the number of training steps, hence existing designs are not a desirable solution.

To address these challenges of privacy leakage when publishing the original training model and the privacy budget, we propose a novel Privacy Preserving Generative Adversial Network (PPGAN) in this work. In the proposed PPGAN based on differential privacy, we inject Laplace noise into the objective function of discriminator to protect the training data.

As we inject Laplace noise into the objective function instead of perturbing the gradients, the privacy budget will therefore not be affected by the amount of generated data and the number of discriminator training steps. Our salient contributions are summarized as follows:

- We propose a *Privacy Preserving Generative Adversarial Network* (PPGAN), where the privacy of training data is guaranteed by the notion of differential privacy.
- To preserve the differential privacy, we leverage the functional mechanism to perturb the objective function in the GAN. Specifically, we first approximate the objective function of discriminator to a polynomial form by utilizing the Taylor Expansion. Then we inject the Laplace noise into the coefficients of objective function. As we leverage perturbed objective function during the training process, the privacy budget becomes unaffected by the cardinality of the training set and also doesn't accumulate in each generator learning step.
- In performance evaluation, we conduct simulations to explore the efficacy of our proposed PPGAN on benchmark dataset MNIST. The simulation results illustrate that our PPGAN is supposed to generate real-like fake images under reasonable privacy guarantee. In addition, comparing to the DPGAN proposed in [80], our proposed PPGAN outperforms significantly.

5.2 Preliminaries

5.2.1 Generative Adversarial Networks

In generative adversarial networks (GANs) [5], two neural networks defined as discriminator (D) and generator (G) are trained simultaneously to model a zero-sum game. The generator aims to map from a random distribution p_z in latent space to the real data distribution p_{data}, hence the results of generator are supposed to be real-like fake samples. While the discriminator is a classifier trying to distinguish whether an input is a real sample or a sample produced by the generator. Based on the feedback from the discriminator, the generator intends to generate better fake samples in order to deceive the discriminator, until it produces closest real-like fake samples. Then, the learning of GAN comprises of finding an optimal parameter y^* for the generator by solving the following minimax objective function,

$$\min_{y} \max_{w} f(y, w) := \mathbb{E}_{x \sim p_{data}}[\log(D_w(x))] + \mathbb{E}_{z \sim p_z}[\log(1 - D_w(G_y(z)))], \tag{8}$$

where w is a discriminator parameter and y is a generator parameter. The output of the discriminator D spans $\{0, 1\}$ i.e., $D_w(x) = 1$, $D_w(x) = 0$, which indicates that the discriminator D classifies the sample x to be a real sample and generated sample respectively. This learning ends with a generator parameter y that best approximates the optimal y^*.

5.2.2 Discriminator learning

The discriminator D updates the parameter w, given a fixed generator G_y. The objective is to find the parameter w which maximizes $f(y, w)$ with a fixed y. The loss function used to learn the discriminator is presented as

$$f_{disc}(L_r, L_g) := \frac{1}{m} \sum_{x \in L_r} \log(D_w(x)) + \frac{1}{m} \sum_{z \in L_g} \log(1 - D_w(G_y(z))), \tag{9}$$

where L_r is a batch of m real data randomly sampled from the training dataset and L_g is a batch of m random vectors generated from the noise distribution p_z. As shown in [5], a gradient ascent optimizer is used to solve the above loss function f_{disc}.

5.2.3 Generator learning

The generator G updates the parameter y, given the new discriminator parameter w. A gradient descent optimizer is used to solve the following generator loss function f_{gen} over

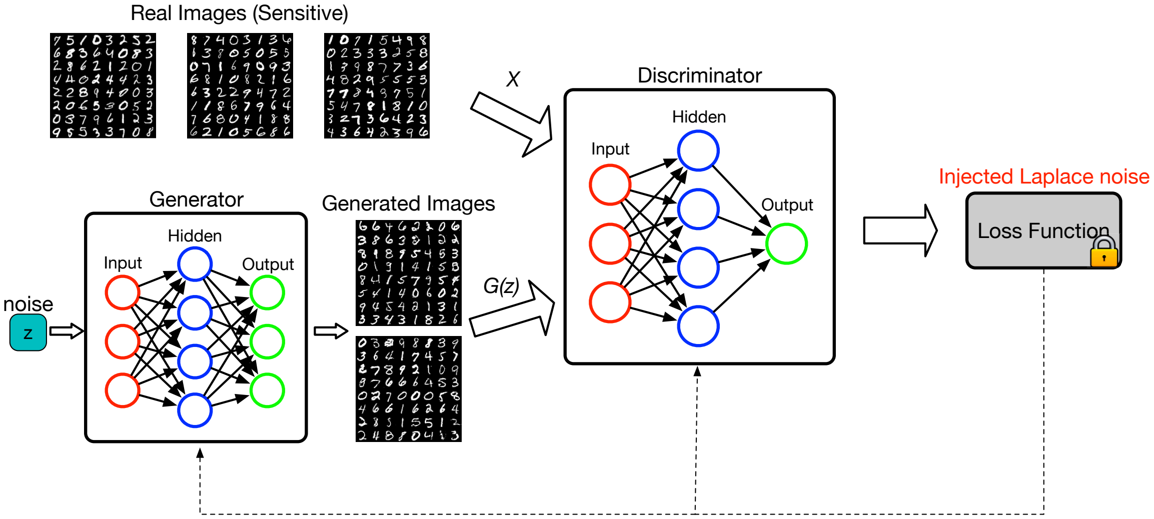


Figure 10: The framework of PPGAN.

the generator parameter y,

$$f_{gen}(L_g) = \frac{1}{m} \sum_{z \in L_g} \log(1 - D_w(G_y(z))), \tag{10}$$

where L_g is a new batch of m random vectors generated from the noise distribution p_z.

5.2.4 Functional Mechanism

Functional mechanism [89] is an extension of the Laplace mechanism that injects differentially private noise into the objective function of regression analysis. Directly publishing the model parameter w in regression analysis compromises the privacy of data [88], due to the relationship among model parameter w, objective function $f(w)$ and data x. In functional mechanism, the objective function $f(w)$ is first represented in polynomial forms through Taylor Expansion and then the Laplace noise is injected into the polynomial coefficients of the expansion, providing a way for privacy preserving data publishing. Consequently through functional mechanism, we can publish the noisy model parameter $\overline{w}$ derived from the noisy objective function $\overline{f}(w)$ without disclosing data privacy.

The model parameter w is a vector containing several values $w_i \dots w_d$. We assume that the product of model parameter values is $\phi(w) = w_1^{c_1} w_2^{c_2} \dots w_d^{c_d}$, where $c_1 \dots c_d \in \mathbb{N}$, and the set of the product $\phi(w)$ is Φ_j, where j is the order of the product $\phi(w)$ from a set $\mathcal{J} = 1, \ldots, j, \ldots, J$. Therefore, the polynomial representation of objective function $f(w)$ can be expressed as

$$f(w) = \sum_{j \in \mathcal{J}} \sum_{\phi \in \Phi_j} \sum_{x \in \mathcal{X}} \lambda_{\phi x} \phi(w), \tag{11}$$

where $\lambda_{\phi x}$ is polynomial coefficients. The Laplace noise is added to polynomial coefficients of the Taylor expansion representation of $f(w)$ as $\overline{\lambda}_\phi = \sum_{x \in \mathcal{X}} \lambda_{\phi x} + \text{Lap}(\frac{\Delta}{\epsilon})$ and the noisy objective function is denoted as

$$\overline{f}(w) = \sum_{j \in \mathcal{J}} \sum_{\phi \in \Phi_j} \overline{\lambda}_\phi \phi(w), \tag{12}$$

where $\Delta = 2 \max_x \sum_{j \in \mathcal{J}} \sum_{\phi \in \Phi_j} \|\lambda_{\phi x}\|_1$ [89]. Therefore, the model parameter $\overline{w}$ that minimizes the noisy objective function $\overline{f}(w)$ satisfies ϵ-differential privacy.

5.3 Problem Description and Methodology

5.3.1 Problem Description

In this work, we suppose that a data provider would like to publish a database $\mathcal{D}$ without compromising the privacy. Specifically, each row in the database $\mathcal{D}$ consists of public variables Y and private variables X ($X \in Y$). The goal is to generate a good representation $\hat{X}$ of X that guarantees differential privacy. To this end, we propose a novel *Privacy Preserving GAN* (PPGAN), a promising solution to learn a differentially private representation $\hat{X}$ with desirable data utility. The architecture of PPGAN is shown in Figure 10. The generator cannot directly access the real data X in the learning process, however, the sensitive information is able to be propagated to the generator through gradients of the discriminator. In order to protect the privacy of training data, in PPGAN, Laplace noise is injected to the loss function of GAN. Consequently, sensitive data X is fed into the discriminator D with a differentially private loss function. This discriminator is used to train a differentially private generator G to generate realistic data samples with privacy guarantee. The main innovation of our work stems from achieving differentially private discriminator learning based on functional mechanism.

Algorithm 3 Privacy Preserving Generative Adversarial Networks (PPGAN)

```
Input: privacy parameters $(\epsilon, \delta)$, batch $L_r$ and $L_g$ with size $m$, total number of training data $M$, number of generator iterations $n_g$, number of discriminator iterations per generation iteration $n_d$, learning rates of discriminator and generator $\eta_d, \eta_g$.
Output: Differentially private generator $G$.
Initialization of discriminator parameter $w$ and generator parameter $y$.
for $n_g$ iterations do
    Take a random batch $L_g$ of noise samples $\{z_i\}_{i=1}^m$ from $p_z$
    Take a random batch $L_r$ of real data samples $\{x_i\}_{i=1}^m$ from $p_{data}$
    for $n_d$ iterations do
        Construct Taylor Expansion of the objective function as (15)
        Inject Laplace noise into the coefficients to construct differentially private loss function $\overline{f}(w)$ (16)
        Compute gradient ascent to update the discriminator:
        $w \leftarrow w + \eta_d \nabla \overline{f}(w)$
    end for
    Take another random batch of noise samples $\{z_i\}_{i=1}^m$ from $p_z$
    Compute gradient descent to update the generator:
    $y \leftarrow y - \eta_g \nabla f(y)$
end for
return $y$
```

5.3.2 Privacy Preserving Generative Adversarial Network

To achieve this goal, we first approximate the loss function of discriminator to polynomial forms by utilizing Taylor Expansion. Then, we inject Laplace noise into polynomial coefficients to reconstruct a perturbed loss function and derive a differentially private discriminator to minimize this perturbed loss function.

Since the discriminator D is a neural network, we apply stacking operation to stack all the hidden layers $\{h_{(1)}, h_{(2)}, \cdots, h_{(k)}\}$ of the discriminator, denoted as h. Thus, we have the following expression of the discriminator loss function,

$$\begin{aligned} f_{disc}(L_r, L_g) &= \frac{1}{m}\sum_{x\in L_r}\log(D_w(x)) + \frac{1}{m}\sum_{z\in L_g}\log(1 - D_w(G_y(z))), \\ &= -\frac{1}{m}\sum_{i=1}^{m}(\log(1+e^{-h_{x_i}w^T}) + \log(1+e^{h_{z_i}w^T})), \\ &= -\frac{1}{m}\sum_{i=1}^{m} F_1(H_1(i,w)) + F_2(H_2(i,w)), \qquad (13) \end{aligned}$$

where x_i is i-th real data samples in batch L_r and z_i is i-th noise samples in batch L_g. Here, we assume $F_1(b) = \log(1+e^{-b})$, $F_2(b) = \log(1+e^{b})$, $H_1(i,w) = h_{x_i}w^T$ and $H_2(i,w) = h_{z_i}w^T$.

Based on Taylor Expansion, we derive the polynomial approximation of $f_{disc}(L_r, L_g)$ as

$$\hat{f}_{disc}(L_r, L_g) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{2}\sum_{k=0}^{\infty}\frac{F_j^{(k)}(a_j)}{k!}(H_j(i,w) - a_j)^k. \qquad (14)$$

Let $a_j = 0$, we can derive $F_1^{(0)}(0) = \log 2$, $F_2^{(0)}(0) = \log 2$, $F_1^{(1)}(0) = -\frac{1}{2}$, $F_2^{(1)}(0) = \frac{1}{2}$, $F_1^{(2)}(0) = \frac{1}{4}$ and $F_2^{(2)}(0) = \frac{1}{4}$. Therefore, Equation 14 can be simplified as

$$\hat{f}_{disc}(L_r, L_g) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=0}^{2}\sum_{x\in\{x_i\}\cup\{z_i\}}\lambda_x^{(k)}(h_x w^T)^k, \qquad (15)$$

where we consider an approximate approach to reduce the degree of the summation, i.e., $k = 0, 1, 2$. Given the above polynomial approximation, we perturb $\hat{f}_{disc}(L_r, L_g)$ by injecting the Laplace noise $\frac{1}{m}\text{Lap}(\frac{\Delta_f}{\epsilon})$ into its polynomial coefficients $\lambda_x^{(k)}$, and the noisy coefficients can be represented as $\overline{\lambda}_x^{(k)}$. With the perturbed coefficients, we can construct the differentially private loss function as

$$\overline{f}_{disc}(L_r, L_g) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=0}^{2}\sum_{x\in\{x_i\}\cup\{z_i\}}\overline{\lambda}_x^{(k)}(h_x w^T)^k. \qquad (16)$$

5.3.3 Privacy Analysis of PPGAN

To validate that the proposed PPGAN indeed provides differential privacy guarantee, we show that the parameters y of the generator G (through the parameters w of the discriminator D) guarantee differential privacy with regard to the sampled training data. In order to prove the privacy guarantees, we provide the sensitivity Δ_f of the polynomial approximation of $f_{disc}(L_r, L_g)$ in the following lemma.

Lemma 5 (**l_1-Sensitivity of objective function**) *Let L_r and L'_r be any two adjacent batches differing in at most one record. Let $\hat{f}_{disc}(L_r, L_g)$ and $\hat{f}_{disc}(L'_r, L_g)$ be the loss functions on L_r and L'_r, then we have the following inequality,*

$$\Delta_f = \sum_{k=0}^{2} \| \sum_{i=1}^{m} \lambda_{x_i}^{(k)} - \sum_{i=1}^{m} \lambda_{x'_i}^{(k)} \|_1 \leq \frac{1}{4}|h|^2 - |h|, \tag{17}$$

where $|h|$ is the number of neurons in last hidden layer.

Proof. Assume that L_r and L'_r differ in the last element x_m and x'_m. We have

$$\Delta_f = \sum_{k=0}^{2} \| \sum_{i=1}^{m} \lambda_{x_i}^{(k)} - \sum_{i=1}^{m} \lambda_{x'_i}^{(k)} \|_1 = \sum_{k=0}^{2} \| \lambda_{x_m}^{(k)} - \lambda_{x'_m}^{(k)} \|_1.$$

We can show that $\lambda_{x_m}^{(0)} = \log 2$ and $\lambda_{x'_m}^{(0)} = \log 2$. Therefore, we have

$$\begin{aligned}
\Delta_f &= \sum_{k=0}^{2} \| \lambda_{x_m}^{(k)} - \lambda_{x'_m}^{(k)}, \|_1 = \sum_{k=1}^{2} \| \lambda_{x_m}^{(k)} - \lambda_{x'_m}^{(k)} \|_1, \\
&\leq \sum_{k=1}^{2} \| \lambda_{x_m}^{(k)} \|_1 + \| \lambda_{x'_m}^{(k)} \|_1, \\
&\leq 2 \max_{x_m} \sum_{k=1}^{2} \| \lambda_{x_m}^{(k)} \|_1, \\
&\leq 2 \max_{x_m} \left(\frac{1}{2} \sum_{e=1}^{|h_{(k)}|} h_{ex_m(k)} + \frac{1}{8} \sum_{e,g} h_{ex_m(k)} h_{gx_m(k)} \right), \\
&\leq \frac{1}{4}|h|^2 + |h|,
\end{aligned}$$

where $h_{ex_m(k)}$ is the state of e-th hidden neuron in $h_{(k)}$. □

Using Lemma 5, we are able to demonstrate that the perturbed loss function $\overline{f}_{disc}(L_r, L_g)$ is ϵ-differentially private in the following lemma.

Lemma 6 *In each generator update, the loss function of discriminator $\overline{f}_{disc}(L_r, L_g)$ satisfies ϵ-differential privacy with respect to the sampled training data.*

Proof. We assume there are two neighboring batches L_r and L'_r, which only differ in the last element x_m and x'_m.

$$\frac{Pr(\overline{f}_{disc}(L_r, L_g))}{Pr(\overline{f}_{disc}(L'_r, L_g))} = \frac{\prod_{k=0}^{2} \exp\,(\frac{\epsilon\| \sum_{i=1}^{m} \lambda_{x_i}^{(k)} - \overline{\lambda}^{(k)} \|_1}{\Delta_f})}{\prod_{k=0}^{2} \exp\,\frac{(\epsilon\| \sum_{i=1}^{m} \lambda_{x'_i}^{(k)} - \overline{\lambda}^{(k)} \|_1}{\Delta_f})}$$

$$\leq \prod_{k=0}^{2} \exp(\frac{\epsilon}{\Delta_f} \| \sum_{i=1}^{m} \lambda_{x_i}^{(k)} - \sum_{i=1}^{m} \lambda_{x'_i}^{(k)} \|_1)$$

$$= \prod_{k=0}^{2} \exp(\frac{\epsilon_f}{\Delta} \| \lambda_{x_m}^{(k)} - \lambda_{x'_m}^{(k)} \|_1)$$

$$= \exp(\frac{\epsilon}{\Delta_f} \sum_{k=0}^{2} \| \lambda_{x_m}^{(k)} - \lambda_{x'_m}^{(k)} \|_1)$$

$$\leq \exp(\frac{\epsilon}{\Delta_f} 2 \max_{x_m} \sum_{k=0}^{2} \| \lambda_{x_m}^{(k)} \|_1)$$

$$\leq \exp(\frac{\epsilon}{\Delta_f} (\frac{1}{4} |h|^2 + |h|))$$

$$= \exp\,(\epsilon)$$

Therefore, the calculation of $\overline{f}_{disc}(L_r, L_g)$ satisfies ϵ-differential privacy. □

Finally this illustrates the privacy guarantee of the parameters y of the generator G.

Theorem 3 *Given the sampling probability $\gamma = \frac{m}{M}$, the number of generator iterations n_g and $\delta' > 0$, the output generator G in Algorithm 3 guarantees (ϵ', δ')-differential privacy with $\epsilon' = 4\gamma^2\epsilon^2 n_g + 2\sqrt{\gamma\epsilon n_g \ln(1/\delta')}$.*

Proof. The proof depends on applying the post-processing theorem in Lemma 7 where the discriminator corresponds to the mechanism $\mathcal{M}$ which takes outputs in $\mathcal{Y}$ (in our case this corresponds to the parameters of the discriminator), and the generator corresponds to the function $\mathcal{F}$ which maps from $\mathcal{Y}$ to $\mathcal{Z}$ (which corresponds to the weights of the generator). Due to the random sampling in Line 6, we ensure the generator is $2\gamma\epsilon$-differentially private by using Lemma 8. To conclude the proof, we apply advanced composition in Lemma 9 with $\epsilon = 2\gamma\epsilon$ and $T = n_g$. □

Lemma 7 (**Post-processing [90]**) *If a mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ is (ϵ, δ)-differentially private and $\mathcal{F} : \mathcal{Y} \to \mathcal{Z}$ is any randomized function, then $\mathcal{F}(M) : \mathcal{X} \to \mathcal{Z}$ is (ϵ, δ)-differentially private*

Lemma 8 (**Privacy amplification via sampling [60]**) *If an algorithm $\mathcal{A}$ is ϵ-differentially private, then for any n-element dataset D, executing $\mathcal{A}$ on uniformly random γn entries ensures $2\gamma\epsilon$-differential privacy.*

Lemma 9 (**Advanced composition [91]**) *Let $\epsilon, \delta' \geq 0$. The class of ϵ-differentially private algorithms satisfies (ϵ', δ')-differential privacy under T-fold adaptive composition for $\epsilon' = T\epsilon^2 + \sqrt{2T\epsilon \ln(1/\delta')}$.*

5.4 Performance Evaluation

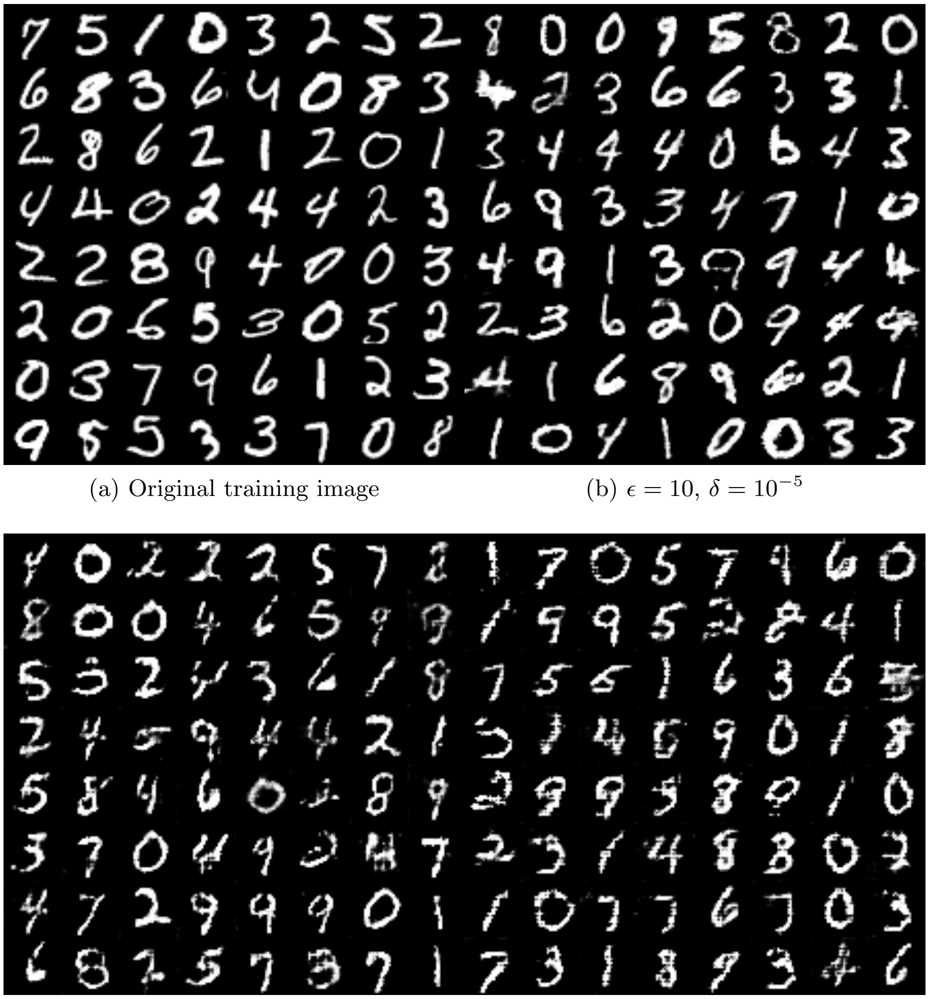
5.4.1 Experiment Settings

In the simulation, we used handwritten digit images from MNIST dataset as the training data to evaluate our PPGAN. There are 60000 training images in MNIST dataset with size of 28 × 28. We train the PPGAN with none labeled training images and map noise to these MNIST digits. The learning rates η_d and η_g are set to 5×10^{-5}. The batch size L is 64, hence the sample rate γ is $64/60000 = 1.1 \times 10^{-3}$. The noise dimension is set to 100. The DCGAN architecture is used in the experiment and the Laplace noise is added to the objective function.

5.4.2 Simulation Results

We firstly investigate the quality of generated images by PPGAN with different ϵ values. The MNIST images shown in Figure 11 are original MNIST image, PPGAN generated images with $\delta = 10^{-5}$, $\epsilon = 10$, 5 and 1 respectively. Here, the privacy budget ϵ is calculated by Theorem 3, where the training steps of generator n_g is set to 25000 in the experiment. Recall that with larger ϵ value, the noise added is smaller and privacy level is lower. From the results shown in Figure 11 with original MNIST image as reference, it is obvious that the proposed PPGAN can generate quite similar images to the training images but not the same images. Moreover, if the privacy budget ϵ is higher, PPGAN can generate much higher quality fidelity fake images. It is evident from Figure 11b that when the value of $\epsilon = 10$, the quality of PPGAN generated image is high. With smaller privacy budget, the generated images are blurrier. Therefore, given a desired privacy budget, PPGAN can generate high fidelity fake images similar to the training images, and doesn't memorize the training images during this generation. In other words, PPGAN can effectively protect the privacy of training data.

Additionally, in order to quantify the quality of generated images, the Inception Score is calculated in the experiment. The Inception Score was first exploited in [92] based on Inception v3 network model. To calculate Inception Score, the input is a list of generated images and the output is the mean and variance score of the list of images. Basically, the result indicates two major characteristics of the generated images. The first determines the diversity of generated images, and the second determines whether the GAN can generate meaningful images. Therefore, a higher Inception Score indicates that the quality of the generated image is higher. The comparison of Inception Score with different GANs is shown in Table 3. It is obvious that with $\epsilon = 10$, the Inception score is high and close to the training images. With decreasing privacy budget, the Inception Score decreases as well. But considering the same privacy budget, it is evident that the inception scores of PPGAN are much higher than DPGAN. Therefore, the proposed PPGAN is effective in generating high quality fake images without compromising the sensitive data during the generation.



(a) Original training image (b) $\epsilon = 10$, $\delta = 10^{-5}$

(c) $\epsilon = 5$, $\delta = 10^{-5}$ (d) $\epsilon = 1$, $\delta = 10^{-5}$

Figure 11: Performance comparison between original training image and generated images with PPGAN.

Table 3: Inception scores with different settings

Dataset	Setting	(ϵ, δ)	Score
MNIST	real	-	1.98 ± 0.29
	DCGAN	-	1.93 ± 0.28
	PPGAN	$(10, 10^{-5})$	1.90 ± 0.30
	PPGAN	$(5, 10^{-5})$	1.84 ± 0.31
	PPGAN	$(1, 10^{-5})$	1.73 ± 0.21
	DPGAN	$(20, 10^{-5})$	1.83 ± 0.42
	DPGAN	$(15, 10^{-5})$	1.65 ± 0.23
	DPGAN	$(10, 10^{-5})$	1.59 ± 0.31

5.5 Conclusion

In our work, we have investigated functional mechanism based differentially private GANs, which can effectively protect the privacy of the published data by adding noise to the coefficients of the objective function in latent space, thereby reducing the overall information loss while guaranteeing privacy. Through extensive simulations, we have shown that the proposed model is reliable to generate real-like synthetic data samples of good quality without disclosing the sensitive information in the training dataset.

6 Location Differentially Private Task Allocation Optimization in Mobile Crowdsensing

6.1 Introduction

With the explosive popularity of the global positioning system (GPS) enabled and various sensors embedded mobile devices such as cellphones and tablets, mobile crowdsensing (MCS) has become a novel sensing and computing paradigm. For instance, the GPS navigation software, Waze, can be considered as a MCS platform, which monitors traffic condition based on the collected data like driving speed from the users. The objective of MCS is to efficiently pair requested tasks and interested workers, involve workers to contribute and analyze data collected by their mobile devices, aggregate and utilize the reported data to reveal information for specific purposes [10]. Therefore, it is obvious that the MCS service has a high demand for workers to sense and gather data from the surrounding environment. There are two types of MCS, participatory crowdsensing and opportunistic crowdsensing, which are classified by different types of participants [93]. The former is that workers participate to fulfill tasks by an incentive, and the latter is that the sensed data collected by mobile device embedded sensors are gathered by the MCS server without the workers' knowledge. Briefly, MCS brings a new method to perceive the world and extends the service of Internet of things (IoT).

In the MCS platform, there are plenty of candidate workers, who are waiting to accomplish tasks. When the task requesters send the task assignments to the MCS server, a subset of appropriate workers will be chosen to participate. Such procedure, called task allocation, is a significant problem that needs to be considered in MCS. Mostly, in task allocation, the workers' travel distance is referred as one of the most common utility metrics. On the workers' side, they prefer to participate in tasks located closer to themselves. On the MCS servers' side, longer travel distance means higher response latency and higher incentive bonus to the workers, which is not desirable [11]. Therefore, based on the travel distance, the tasks can be assigned to workers properly, hence enhancing the efficiency of the MCS. As the MCS requires workers' location, the geocoding system can be considered to represent workers' spatial information, since it can provide the ideal visualization, which is helpful for the MCS server to make decisions for task allocation. For instance, geohash is a public domain geocoding system, which can encode location into a short string of digits by dividing the world map into grids. With geohash, locations can be stored in a database with only one index instead of traditional two joint indexes (i.e., latitude and longitude). The queries for one index are much faster and more efficient than these for two joint indexes.

The normal MCS assumes that the workers' location should be reported to the server in order to provide optimal task allocation by pairing tasks and adjacent workers. Nevertheless, it also raises serious privacy concerns, especially when there is a dishonest MCS server. Furthermore, in task allocation procedure, only a subset of participatory workers are assigned tasks owing to the travel distance. There is no compensation for the location information disclosure of the workers who are not selected for any tasks. Even though the selected workers can obtain an incentive bonus, it cannot offset the privacy leakage. With the exposure of locations, the dishonest server and the malicious party could learn workers' living habits and life patterns, which makes the workers vulnerable to various attacks, even some serious physical attacks. For instance, in August 2018, a violent robbery happened in Maryland. The victim is picked up by a fake Uber driver and the driver demanded the victim's wallet and cellphone with a handgun. Instead of robbery, more serious crimes targeting particular victims, such as kidnapping, sexual assault, etc, might happen because of the leakage of location information, since malicious party could learn people's life pattern through the disclosed locations. Furthermore, as the MCS aims to make MCS workers complete requested tasks efficiently and effectively, it also requires satisfactory service quality. However, it is always challenging to have efficient task allocation, while preserving user's privacy [11, 94].

To address those issues, it is worthy to protect the location data from the dishonest MCS service providers or eavesdroppers. To thwart the dishonest service provider or eavesdropper, we propose two novel differentially private geocoding (DPG) schemes inspired by

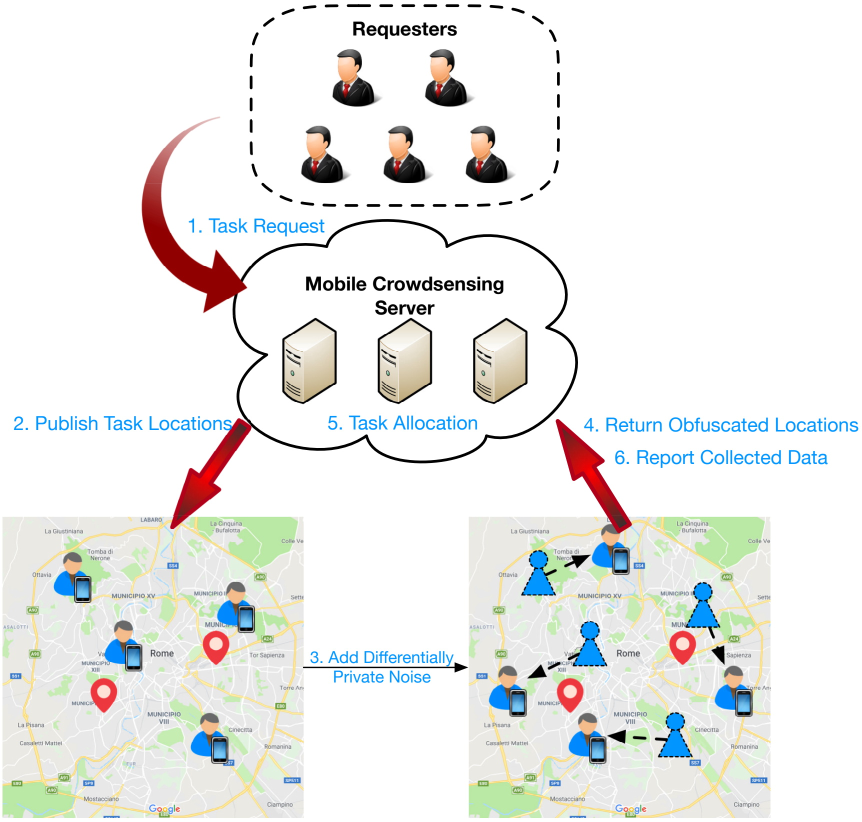

Requesters
1. Task Request
Mobile Crowdsensing Server
2. Publish Task Locations
5. Task Allocation
4. Return Obfuscated Locations
6. Report Collected Data
3. Add Differentially Private Noise

Figure 12: System overview of the MCS task allocation scheme.

gray code and Geohash system to perturb workers' locations. Moreover, we develop the optimal MCS task allocation based on the obfuscated workers' locations, while minimizing the total travel distance of all workers. However, with the DPG location privacy preservation mechanism, a worker's location is perturbed by differential noise. Hence, there comes a challenge that the travel distance of each worker is uncertain now due to the introduced differential noises. Aiming to address those challenges, we leverage the conditional value at risk (CVaR) [95] to characterize the travel distance uncertainty brought by the DPG scheme. Then, we provide optimal MCS task allocation with consideration of quantitatively analyzing the service quality by minimizing the total travel distance under the uncertainty, while preserving MCS workers' location privacy by differential privacy settings. Our salient contributions are summarized as follows:

- In order to optimize task allocation in MCS while protecting the workers' location privacy, we formulate the travel distance minimization problem based on the workers' obfuscated locations. On the MCS workers' side, they are trying to fulfill their interested tasks without disclosing the location privacy. On the MCS server's side, the target is to make the best use of the MCS workers' obfuscated location data, and meanwhile effectively allocate the mobile crowdsensing tasks.

- We propose two novel differentially private geocoding (DPG) mechanisms based on local differential privacy in geocoding system according to Geohash system and gray code. We first encode the workers' location data with predefined geocoding system. In order to improve the utility, the geocode of each grid only differs by one single bit with the horizontal and vertical adjacent grids and differs by two bits with the diagonal adjacent grids. Then, the ϵ-differentially private noise is injected to the geocode of each worker to protect workers' location privacy.

- The formulated travel distance minimization problem is basically an integer linear programming (ILP) problem. However, it is difficult to solve the problem, because the distance between workers and task locations is a random variable due to the injected differentially private noise. Therefore, we develop a heuristic algorithm based on CVaR to give a feasible solution to the optimization problem.

- Based on the real-world datasets for taxi cabs in Rome, we conduct extensive simulations to verify the effectiveness of the proposed location differential privacy preserving MCS scheme.

6.2 Related Work

As a novel sensing and computing paradigm, MCS gained tremendous popularity [96, 11, 97]. In [96], the authors design two incentive mechanisms for MCS based on Stackelberg game and reverse auction. Therefore, the utility is maximized and the computational complexity is reduced in the reverse auction-based incentive mechanism. The authors in [11], formulated the task allocation optimization problem into a mixed-integer linear-programming problem and provided feasible solution. Similar to task allocation problem, in ridesharing, when pairing the drivers and passengers, the traveling distance is supposed to be small too. For instance, in [98], the authors deploy differential privacy to protect users' location privacy while minimizing the total cruising distance of ridesharing drivers during the scheduling. Compared to this work, we leverage the CVaR and give a feasible solution to our formulated traveling distance minimization problem under the characterization of uncertainty, which is brought by differentially private noise.

With the proliferation of location-based services, many works focus on location privacy preservation schemes recently, where the service provider is considered to be honest-but-curious. The location protection approaches can be divided into three categories. The first one is obfuscation-based method, which is also called dummy-based method [99, 100]. In this case, users send dummy requests together with the true request, and hence the attacker cannot distinguish the real location from the dummy locations. In [101], the authors proposed a dummy location generation scheme in location-based services by reported movement locations from users. However, this kind of approaches will compromise the quality of service because of the inaccuracy of the location data and the computational complexity is pretty high. The second category is collaboration-based methods, where each user sends a time or space obscure cloak region to the servers instead of the true location [102, 103, 104]. In [102], the authors proposed a user-collaborative privacy-preserving approach that LBS users can seek information directly from their nearby peers and when the users cannot obtain the information from peers, they would query the LBS. Generally speaking, in collaboration-based methods, it may need additional high cost preprocessing of the data which may further incur high communication cost. The last kind of methods is identity and location anonymity. The mix-zone model [105, 106] is first proposed to be used in location privacy preservation in [107]. A mix-zone indicates that when users enter the mix-zone, they can change their pseudonym to prevent adversary from tracking their locations. Furthermore, some schemes belonging to this category often put the true location together with another $k-1$ dummy locations in an area from historical data or other users' location to guarantee k-anonymity [108]. The main concern in this category is that a trustworthy anonymizer is required to construct the anonymity settings, nevertheless the anonymizer can also be the malicious party.

6.3 Network Model and Preliminaries

In this section, we describe our network model for MCS task allocation, and introduce the differential privacy preliminaries and the applications in location privacy.

6.3.1 System Description

In MCS, there consists of three parts, the MCS server, task requesters and workers. Mobile crowdsensing is a technique that the server is capable to aggregate and analyze data collected by workers' sensor and GPS embedded mobile devices in order to reveal information for some specific tasks. In our work, in order to protect the MCS users' location privacy, a differential privacy based location privacy preservation scheme, called DPG, is designed, which will be introduced in Section 6.4.1. As the worker's location is obfuscated by the DPG scheme, the true travel distance between the worker and the task is uncertain. Therefore, CVaR is employed to characterize the distance uncertainty.

As shown in Figure 12, in our architecture, a number of requesters will first send task requests to the MCS server. For example, city government or other organizations can be a task requester that asks to monitor air quality of a specific area. Next, after the server receives the request, it will publish the task on the MCS platform. The workers who are interested in the task will share their location information with the MCS server. However, as talked in Section 6.1, the workers' location privacy might be compromised during this step as the malicious parties have many opportunities to obtain the workers' location data, and especially sometimes the MCS server could be dishonest. Therefore, in our work, the workers will only send the obfuscated location generated by applying a differential privacy based location privacy preservation mechanism to the MCS server. The MCS server will generate a geocoding map for a specific area, such as the map shown in Figure 13, and send the map to each worker. Then, the workers will encode their location with the geocoding system and inject differentially private noise to the encoded location. According to the obfuscated locations reported by the workers, the MCS server will allocate requested tasks to interested workers in accordance with the distance between the workers' obfuscated locations and the location of the task. The MCS server then can formulate a travel distance minimization problem and find the optimal task allocation solution. We assume that the set of workers is $\mathcal{W} = \{w_1, \cdots, w_i, \cdots, w_n\}$ with the size of n. The set of the task can be represented with $\mathcal{T} = \{t_1, \cdots, t_j, \cdots, t_m\}$ with the size of m. The location of a worker and a task can be denoted as L_{w_i} and L_{t_j}, and the obfuscated locations are L'_{w_i} and L'_{t_j} respectively. The distance between a worker and a task is expressed as $d(w_i, t_j)$.

6.3.2 Threat Model

During the MCS task allocation, the server will pair workers with tasks. Our target is to protect workers' location privacy during the task allocation procedure since the MCS server can obtain workers' location information continuously before they accept tasks. Moreover, the workers who are not assigned any tasks are also required to share their location information and the privacy sacrifice cannot be compensated. We assume the attackers want to learn workers' private location data and either the MCS server, a participatory worker or the third party identity can be considered as attackers. However, data pollution attacks, that malicious workers would modify their location and try to affect the overall task allocation results, are beyond the scope of this work. We suppose that the attackers are able to obtain side information or arbitrary background knowledge of workers. Our objective is to hide the workers' true location despite the prior knowledge of adversaries.

6.3.3 Location Differential Privacy Preliminaries

In centralized DP, it is assumed that the database or data aggregator is trustworthy. Therefore, local differential privacy (LDP) is defined without relying on a trustworthy third party. LDP is basically developed on the Warner's randomization response model [45].

Similar to the centralized DP, the definition of *local differential privacy* is shown as follows.

Definition 6.1 (Local Differential Privacy [109]) With a privacy confidence parameter $\epsilon \geq 0$, a randomized algorithm $\mathcal{A}$ satisfies ϵ-local differential privacy, when given two inputs x and x',

$$\frac{Pr[\mathcal{A}(x) = z]}{Pr[\mathcal{A}(x') = z]} \leq e^{\epsilon},$$

where z is the secure view of the input.

Definition 6.2 (Local d-Privacy [110]) With a privacy confidence parameter $\epsilon \geq 0$, a randomized algorithm $\mathcal{A}$ satisfies local d-privacy, when given two inputs x and x',

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

where z is the secure view of the input and $d(\cdot, \cdot)$ is a distance metric.

In this definition, the distance metric $d(\cdot, \cdot)$ is supposed to satisfy three properties: $d(x, x) = 0$, $d(x, x') = d(x', x)$ and $d(x, x') + d(x', x'') \geq d(x, x''), \forall x, x', x''$. Based on different practical scenarios, we can choose different distance metrics. For instance, in location-based systems, Euclidean distance is suitable, while in database applications, hamming distance is suitable. The well-known location differentially private scheme, geo-indistinguishability [111, 112], is defined based on the local d-privacy concept, where Euclidean distance is used as the distance metric. The geo-indistinguishability scheme can protect location privacy in despite of side information known by the attackers.

6.4 Travel Distance Minimization Problem and Differentially Private Geocoding Scheme

In this section, we propose two novel DPG location privacy preservation mechanisms and formulate the travel distance minimization problem with MCS workers' obfuscated location data. We give a feasible solution to the problem by exploiting CVaR to characterize the uncertainty of workers' location.

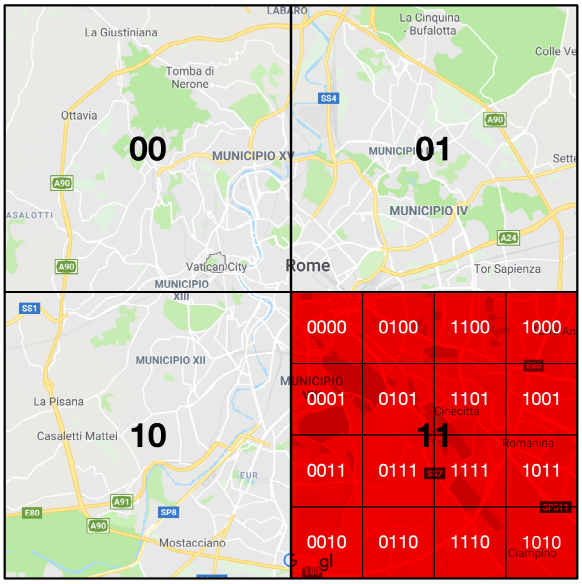


Figure 13: The DPG coding example.

6.4.1 Geocoding Based Location Differential Privacy Preserving Schemes

For the traditional differential privacy setting, a trusted third party database is required. However, in reality, sometimes even some MCS servers are not trust-worthy. Therefore, we leverage local differential privacy to protect each worker's location privacy. Intuitively, each worker can inject DP noises to his/her true location locally and report the obfuscated location to the MCS server, hence the MCS server is not able to access to the worker's true location. Moreover, geocodes can provide a proper method for the MCS server to check the eligibility of task assignments by a clear visualization. Consequently, the MCS server can quickly inform and make decisions for task allocation. Therefore, in our work, we propose two differentially private geocoding (DPG) schemes inspired by gray code and Geohash system based on locally differential privacy (LDP) definition without relying on a trust-worthy third party.

6.4.2 Gray code DPG

For the gray code DPG, rooting from the gray code, the targeted area can be first divided into a number of macro grids. Then, each macro grid is subdivided into several micro grids. All the geographical points in one grid can be grouped and encoded by a code with a string of digits. Each code has a prefix with fixed length to represent the macro grid and a suffix with four digits to represent the located micro grid in which the location lies. For example, as shown in Figure 13, the map is first divided into four macro grids and each macro grid is subdivided into sixteen micro grids like the red region. The worker encodes his/her location according to the geocoding map into a code c. For instance, the red micro grid at the right bottom corner can be encoded with 111010, where the prefix of 11 indicates that the grid is belonging to the bottom right macro grid and the suffix of 1010 marks that the grid is located at the southeast corner micro grid. The MCS server is able to control the size and quantity of macro grids for the targeted area with different length of the prefix. If the MCS server requires higher resolution of the divided grid, it can further divide the area with a larger number of macro grids with a longer prefix. Moreover, we assume the MCS server is responsible for generating this geocoding system for the workers. In general, the worker will first download the generated geocoding map from the MCS server and encode his/her true location with such geocoding system into a micro grid. As we are using geocoding system to present physical world locations, we need to consider the proximity problem. Therefore, inspired by gray code, in our geocoding system, the code of each micro grid only differs by one bit with the horizontal and vertical neighboring micro grids, and differs by two bits with the diagonal neighboring micro grids, which can also represent the distance in the physical world.

To preserve the privacy of the encoded locations, we randomize the code of each location by injecting differentially private noises. In gray DPG scheme, we only perturb the suffix code of each worker and keep the prefix code the same in order to enhance the utility of the data. The code of the worker's located micro grid will be perturbed by injecting locally differentially private noises. Based on the gray code DPG scheme, the worker's obfuscated micro grid is supposed to be in the same macro grid as the true micro grid that he/she is belonging to.

We assume the suffix of each grid can be represented with s, a binary string with the length of g ($g \in \{2^0, 2^1, 2^2, \dots\}$). For instance, we assume $g = 4$, then one suffix can be denoted as $s = [1, 0, 1, 0]$. Each perturbed bit of the suffix can be defined as $s'[k](k \in [g])$. The probability of turning the real suffix bit $s[k]$ to the obfuscated suffix bit $s'[k]$ is defined as:

$$\forall_{k\in[g]} P(s'[k] = 0) = \begin{cases} p = \frac{e^{\frac{\epsilon}{g}}}{e^{\frac{\epsilon}{g}}+1}, \text{when } s[k] = 0, \\ \\ q = \frac{1}{e^{\frac{\epsilon}{g}}+1}, \text{when } s[k] = 1, \end{cases} \tag{18}$$

$$\forall_{k\in[g]} P(s'[k] = 1) = \begin{cases} p = \frac{e^{\frac{\epsilon}{g}}}{e^{\frac{\epsilon}{g}}+1}, \text{when } s[k] = 1, \\ \\ q = \frac{1}{e^{\frac{\epsilon}{g}}+1}, \text{when } s[k] = 0. \end{cases} \tag{19}$$

In other words, the probability of bit flipping is $\frac{1}{e^{\frac{\epsilon}{g}}+1}$ for a g-bit suffix.

Theorem 4 *With the privacy budget $\epsilon \geq 0$, for any two location codes c_1 and c_2 with the same prefix and any two suffixes s_1 and s_2, the gray DPG scheme satisfies ϵ-local differential privacy.*

Proof. For any two location codes c_1 and c_2 with two suffix codes s_1 and s_2 including four digits and an obfuscated location code z, we have

$$\begin{aligned} \frac{P[\mathcal{A}(c_1) = z]}{P[\mathcal{A}(c_2) = z]} &= \frac{P[\mathcal{A}(s_1) = z]}{P[\mathcal{A}(s_2) = z]} = \frac{P(z \mid s_1)}{P(z \mid s_2)} \\ &= \frac{\Pi_{i=0}^{g} P(z[i] \mid s_1[i])}{\Pi_{i=0}^{g} P(z[i] \mid s_2[i])} \\ &\leq \left(\frac{P(z[0] \mid s_1[0])}{P(z[0] \mid s_2[0])}\right)^g \\ &\leq \left(\frac{p}{q}\right)^g = e^{\epsilon}. \end{aligned}$$

Therefore, it is satisfying the definition of local differential privacy. □

6.4.3 Geohash DPG

The other DPG that we propose is based on the Geohash system, where each location can be encoded with a string of binary digits. Similar to the gray code DPG, as shown in Figure 13, the MCS server can generate the geocoding map by dividing the area into small grids, give a code c to each grid and share the geocoding map to all participated workers. Then, the workers are able to encode their location into the Geohash code. In order to protect the workers' location privacy, the workers can obscure the true locations within a range of some grids. We assume that each worker's true location is hidden in h grids, h is within the set of $\{2^0, 2^1, 2^2, \dots\}$ and h is supposed to smaller than the total number of divided grids. The MCS server will choose the value of h and send to all workers. We can set h as the radix to represent the Geohash code. For instance, a Geohash code can be expressed as $(011011)_2$ in binary system with base 2 and the same code can be represented as $(1\text{B})_{16}$ in hexadecimal system with radix 16. Then, the last digit $c[v]$ of the Geohash code can be perturbed by the following probability,

$$\forall_{a \in [h]} P(c'[v] = a) = \begin{cases} p = \frac{e^\epsilon}{e^\epsilon + h - 1}, \text{when } c[v] = a, \\ \\ q = \frac{1}{e^\epsilon + h - 1}, \text{when } c[v] \neq a. \end{cases} \tag{20}$$

Theorem 5 *With the privacy budget $\epsilon \geq 0$, for any two Geohash codes c_1 and c_2 with the same prefix and the last bits for them are $c_1[v]$ and $c_2[v]$, the Geohash DPG scheme satisfies ϵ-local differential privacy.*

Proof. For any two Geohash codes c_1 and c_2 with the same prefix and an obfuscated Geohash code z, we have

$$\frac{P[\mathcal{A}(c_1) = z]}{P[\mathcal{A}(c_2) = z]} = \frac{P(z \mid c_1)}{P(z \mid c_2)} = \frac{P(z[v] \mid c_1[v])}{P(z[v] \mid c_2[v])} \leq \left(\frac{p}{q}\right) = e^\epsilon.$$

Therefore, it satisfies the definition of local differential privacy. □

After randomization, the worker will send the obfuscated code back to the MCS server.

Then, according to the obfuscated code, the MCS server will optimize the task allocation, which is discussed in Subsection 6.4.4.

6.4.4 MCS Travel Distance Minimization with Workers' Location Privacy Preservation

So far, the DPG scheme is introduced and hence in this subsection, we will give the formulation of MCS travel distance minimization problem with location privacy preservation.

As we described before, the MCS workers' location data is obfuscated by the DPG scheme. With the obscure location data, the distance $d(w_i, t_j)$ between each worker's real location L_{w_i} and the location of a task L_{t_j} is uncertain. The sanitized distance between L'_{w_i} and L'_{t_j} is denoted by $\tilde{d}(w_i, t_j)$. According to the sanitized distance, the MCS server could make decisions to pair workers and tasks efficiently and effectively. In order to select workers for a particular task, we denote

$$x_{ij} = \begin{cases} 1, & \text{if worker } w_i \text{ is chosen for task } t_j, \\ 0, & \text{otherwise.} \end{cases} \tag{21}$$

Consenquently, the travel distance minimization problem for MCS can be formulated as follows,

$$\min_{x} \quad \sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}\tilde{d}(w_i, t_j), \tag{22}$$

s.t.:

$$x_{ij} = \{1, 0\}, \tag{23}$$

$$\sum_{j=1}^{m} x_{ij} \leq 1, \quad \forall i, \tag{24}$$

$$\sum_{i=1}^{n} x_{ij} = 1, \quad \forall j. \tag{25}$$

In the formulation, the integer x_{ij} is the optimization variable. The inequality (24) indicates that each worker cannot be assigned more than one task. The equation (25) shows that each task must be finished by a worker. The formulated travel distance minimization problem is an integer linear programming (ILP) problem if the travel distance $\tilde{d}(w_i, t_j)$ is fixed. However, due to the DPG scheme, the obfuscated distance $\tilde{d}(w_i, t_j)$ is uncertain, which makes the optimization problem difficult to be solved. Therefore, we utilize CVaR to characterize the distance uncertainty and then solve the converted ILP problem.

6.4.5 Solutions to the Proposed Optimization

In order to evaluate the uncertainty, we leverage conditional value at risk (CVaR) [95] to achieve high reliability. By incorporating the CVaR into the proposed optimization problem, the risk brought by the uncertainty can be minimized. Moreover, β-CVaR can be defined as the conditional expectation of travel distance not exceeding the amount α with a probability level of β. Normally, β is also called confidence level and can be set as 90%, 95%, and 99%. In our problem, since the obfuscated location for each worker is generated by equation (18), (19), and (20), we can construct the travel distance distribution in $\mathbb{R}$ of each worker. Because the travel distance is not normally distributed, it is difficult to apply value at risk (VaR) approach to solve the problem. We assume the total travel distance can be defined as $f(x, \tilde{d}) = \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} \tilde{d}(w_i, t_j)$. In our formulation, x_{ij} is the decision parameter and $\tilde{d}(w_i, t_j)$ is the random variable with density $p(d)$, so $f(x, \tilde{d})$ is also a random variable caused by the distribution of $\tilde{d}(w_i, t_j)$. Consequently, VaR_β and CVaR_β for the travel distance minimization problem can be written as

$$\alpha_\beta(x) = \min\{\alpha \in \mathbb{R} : \int_{f(x,\tilde{d}) \leq \alpha} p(d) \mathrm{d}d \geq \beta\} \tag{26}$$

and

$$\phi_\beta(x) = \frac{1}{1-\beta} \int_{f(x,\tilde{d}) \geq \alpha_\beta(x)} f(x, \tilde{d}) p(d) \mathrm{d}d. \tag{27}$$

The equation (27) indicates that the probability of $f(x, \tilde{d}) \geq \alpha_\beta(x)$ is equal to $1 - \beta$. According to the distribution of $\tilde{d}$, we can define a function F_β as

$$F_\beta(x, \alpha) = \alpha + \frac{1}{1-\beta}\mathbb{E}[f(x, \tilde{d}) - \alpha]^+, \tag{28}$$

where $[y]^+ = \max\{y, 0\}$ and $\mathbb{E}$ is the expected value respect to the density $p(d)$.

Lemma 10 *The CVaR$_\beta$ of the optimization problem with the decision parameter x can be determined as*

$$\phi_\beta(x) = \min_{\alpha \in \mathbb{R}} F_\beta(x, \alpha). \tag{29}$$

The defined function $F_\beta(x, \alpha)$ is convex and continuously differentiable with the parameter α. [95]

Lemma 11 *The minimization of CVaR$_\beta$ with respect to x is equivalent to the minimization of the defined function $F_\beta(x, \alpha)$ of all combination of (x, α), which can be expressed as [95]*

$$\min_x \phi_\beta(x) = \min_{(x,\alpha)} F_\beta(x, \alpha). \tag{30}$$

In other words, a pair (x^*, α^*) achieves the minimum of the defined function $F_\beta(x, \alpha)$ if and only if x^* achieves the minimum of CVaR$_\beta$. As the travel distance minimization problem is convex, the CVaR$_\beta$ with the parameter x is convex and the defined function $F_\beta(x, \alpha)$ is also convex with respect to (x, α).

Moreover, the expected value of CVaR can be approximated by sampling. The samples $d_1, \ldots, d_k, \ldots, d_q$ are generated according to the travel distance distribution $\tilde{d}$ based on the density $p(d)$. The approximation of CVaR can be represented as

$$\tilde{\text{CVaR}}_\beta(x, \alpha) = \min_\alpha \alpha + \frac{1}{q(1-\beta)} \sum_{k=1}^{q} [f(x, d_k) - \alpha]^+. \tag{31}$$

Moreover, the approximation (31) is convex with respect to $(x, \tilde{d})$.

With CVaR, the fat-tail events can be detected in the total travel distance distribution. In addition, CVaR inherits superior properties. For instance, the formulated travel distance minimization problem is an ILP problem, the converted optimization problem based on CVaR is still an ILP problem. Accordingly, we can reformulate our problem with CVaR as

$$\min_{(x,\alpha)} \quad \alpha + \frac{1}{q(1-\beta)} \sum_{k=1}^{q} [f(x, d_k) - \alpha]^+, \tag{32}$$

s.t.:

$$x_{ij} = \{1, 0\}, \tag{33}$$

$$\sum_{j=1}^{m} x_{ij} \leq 1, \quad \forall i, \tag{34}$$

$$\sum_{i=1}^{n} x_{ij} = 1, \quad \forall j, \tag{35}$$

$$\alpha \in \mathbb{R}. \tag{36}$$

Due to the presence of the integer parameter x_{ij}, any feasible solution to the reformulated problem comes from a subset of vertices. Therefore, to solve the ILP problem is equal to find out the minimum vertex cover, which is a typical NP-hard problem [113, 114]. In order to reduce the complexity, we can first relax the binary variable $x_{ij} \in \{0, 1\}$ to real numbers in $[0, 1]$ [115]. With the relaxed optimization variables, we can calculate a lower bound for the formulated problem as the benchmark. Therefore, the ILP problem is converted into the LP problem. It is obvious that the complexity is reduced and the problem can be solved in polynomial time.

The overall procedure of the relax-and-fix heuristic algorithm is shown in Algorithm 4. The binary variables x_{ij} are first relaxed to decimal values between 0 and 1. Accordingly, the ILP problem is converted to an LP problem. We can solve the LP problem and obtain x_{ij} values between 0 and 1. A set $\mathcal{X}_{ij}$ can be constructed consisting of all the x_{ij} values. If there is not any x_{ij} larger than 0.5, we can set the minimum x_{ij} to 0. Otherwise, we will fix the maximum x_{ij} to 1. The fixed x_{ij} will be taken out of the set $\mathcal{X}_{ij}$. Next, we can find the feasible solution for the problem again with the fixed x_{ij} and the remaining relaxed

Algorithm 4 Relex-and-Fix Heuristic Algorithm

```
Input: $x_{ij}$ LP feasible values.
Output: fixed $x_{ij}$-values.
$\mathcal{X}_{ij} \leftarrow$ set of all $x_{ij}$ with decimal values;
while $\mathcal{X}_{ij} \neq \emptyset$ do
    if all fractional values in $\mathcal{X}_{ij} < 0.5$ then
        fix the minimum $x_{ij}$ to 0;
        $\mathcal{X}_{ij} \setminus x_{ij}$;
        reformulate and solve the new relaxed LP problem with fixed $x$-variables;
    else
        fix the maximum $x_{ij}$ to 1;
        $\mathcal{X}_{ij} \setminus x_{ij}$;
        reformulate and solve the new relaxed LP problem with fixed $x$-variables;
    end if
end while
return all fixed $x_{ij}$-values;
```

variables. Similarly, we can fix all the x_{ij} values by iterations until the set $\mathcal{X}_{ij}$ is empty. So far we can feasibly solve the travel distance minimization problem and obtain all fixed integer values of x_{ij}.

The computational complexity of solving the optimization problem is extremely high due to the binary parameter x_{ij}. The possible combinations of x_{ij} reaches to 2^{nm}. As discussed before, after relaxing the binary parameter x_{ij}, the ILP can be converted to an LP problem. In [116], the intrinsic computational complexity of an LP problem is $O(A^3 \cdot L)$. Here, L is the storage cost for the data of the proposed problem, and A is the larger value of the number of constraints and the number variables in the problem. In the relaxed problem, we will have the number of mn variables. Therefore, the computational complexity of solving the LP problem with the number of mn variables is equal to $O((mn)^3 \cdot L)$. Consequently, the computational complexity of solving the original problem can be represented as $O(2^{nm} \cdot (mn)^3 \cdot L)$. As we relax and fix each binary variable x_{ij} by iterations, the complexity for the iteration is $O(mn)$, hence the total computational complexity for the heuristic algorithm is $O((mn)^4 \cdot L)$, which is smaller than that of the original problem.

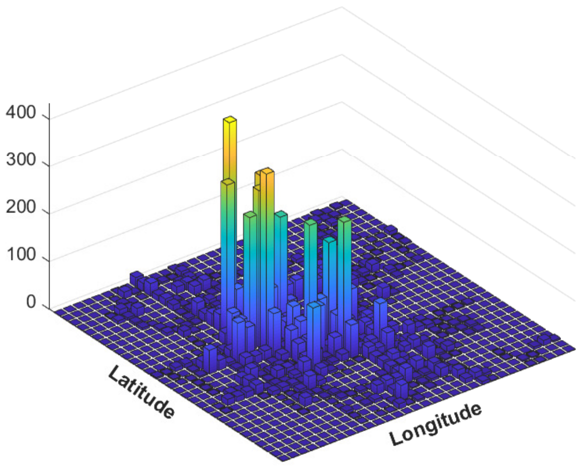


Figure 14: Taxi density at downtown of Rome.

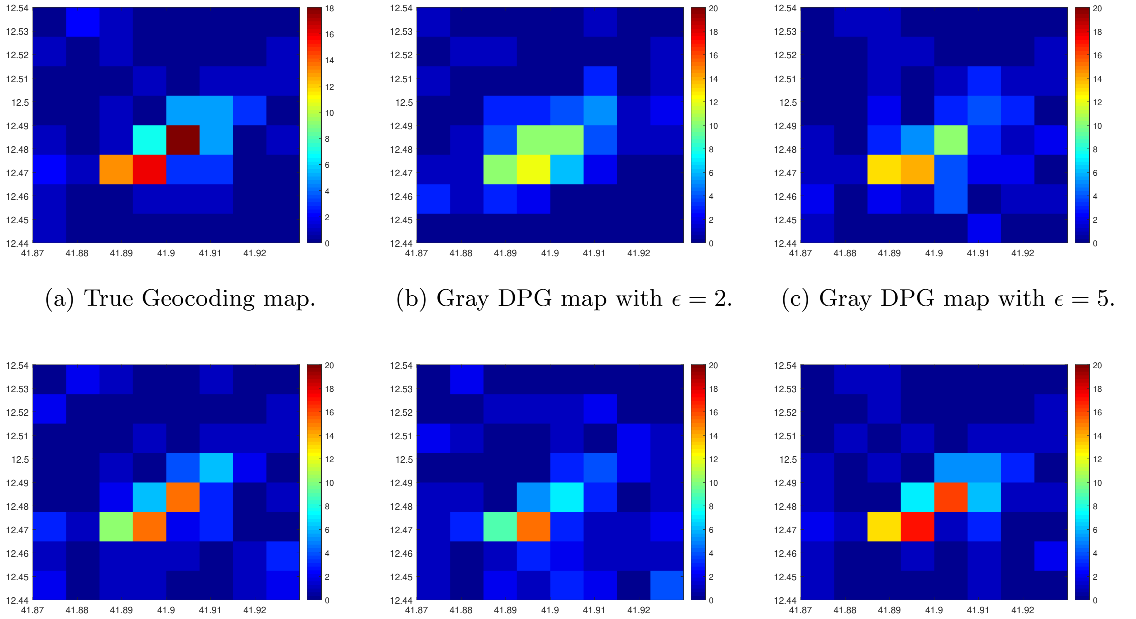
6.5 Simulation Results

In this section, we demonstrate our simulation results of the proposed gray DPG and Geohash DPG on a real-world taxi cab dataset. We first give a clear visualization and evaluate the performance of the proposed schemes. Next, we investigate the impact of different parameters in the given feasible solutions. We employ the geo-indistinguishability mechanism, which is introduced in Section 6.3.3, as the benchmark and give a comparison of the results of the travel distance minimization problem. We conduct the simulations by Matlab with Gurobi optimization solver on a PC with Intel Core i7 CPU and 16.0 GB memory size.

6.5.1 Dataset and Simulation Setup

6.5.2 Dataset

All the simulation results are based on the public location dataset of taxi, which comprises the trajectories of taxi cabs in Rome [117]. Every taxi driver periodically records its GPS locations (latitude and longitude) and sends it with its driver ID to a central server. The total dataset contains the recorded locations about 320 taxi cabs collected over one



(a) True Geocoding map. (b) Gray DPG map with $\epsilon = 2$. (c) Gray DPG map with $\epsilon = 5$.

(d) Geohash DPG map with $\epsilon = 2$. (e) Geohash DPG map with $\epsilon = 4$. (f) Geohash DPG map with $\epsilon = 5$.

Figure 15: Geocoding map of the two DPG schemes.

month. The density of all taxis' recorded locations in Figure 14 show the unevenly distribution of taxi cabs over the downtown area of the city. The GPS coordinate of each location sample can be mapped into the region (i.e., the macro grid and micro grid) it falls into based on the geocode map. It is obvious that most of the taxi cabs are at the center of the downtown area. In the simulation, we assume that all the tasks are located within the latitude (41.87, 42.93) and longitude (12.44, 12.54). In the simulation, for convenience, we calculate the distance between any two locations with kilometers instead of longitude and latitude.

6.5.3 Simulation Setup

In the simulations, we randomly select 100 workers from the dataset and the task locations are generated uniformly in the specific area. Each worker can only apply for one task that is closest to him/her. We investigate the impacts of several key parameters when solving the travel distance minimization problem. The number of task assignments m is one

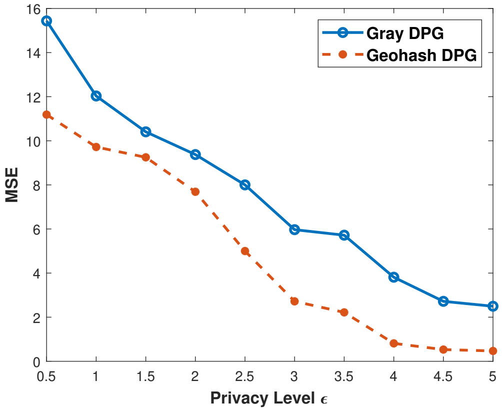


Figure 16: Impact of DP level ϵ to the accuracy of the density map.

of the key paramters. In the two proposed DPG schemes, the privacy level ϵ is the significant parameter. In the literature, typically we can choose ϵ between 0.1 and 10 [118, 119]. The CVaR is employed to characterize the uncertainty brought by the differentially private noises. There are two important paramters in CVaR, the confidence level β and the number of samples q. The default values for these parameters are $m = 10$, $\epsilon = 5$, $\beta = 0.9$ and $q = 500$.

The evaluation metric for the task allocation efficiency is the Mean Travel Distance (MTD) of the selected workers and their assigned task locations

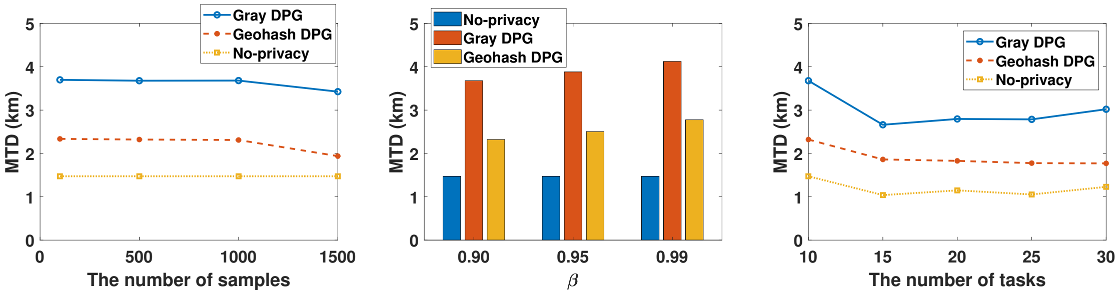
$$\text{MTD} = \frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij} d(w_i, t_j), \tag{37}$$

where n is the number of workers, m is the number of task assignments, and $d(w_i, t_j)$ is the distance between selected worker w_i and the task t_j.

We compare our mechanism **DPG w/ CVaR** with other two benchmarks.

- **No-Privacy**: The worker selection mechanism when each worker reports its real location to the MCS sever.

- **Geo-indistinguishability (Geo-I) mechanism**: As introduced in Section 6.3.3, the high dimensional Laplace noise is injected to the worker's true location. To compare the performance, we use Geo-I to protect worker's location privacy and solve the same



(a) The number of samples.

(b) β value.

(c) The number of tasks.

Figure 17: Comparison results with different parameters.

optimization problem illustrated in Section 6.4.4 with CVaR.

6.5.4 Performance Evaluation

We first investigate the workers density map of the downtown area at Rome. The generated density map with true locations is shown in Figure 15a. It is obvious that the geocoding system can provide a clear visualization for the MCS server to check how many workers are in a specific grid. The density map is generated by a number of 100 locations randomly selected from the overall dataset. To simplify the simulation, we divide the area into 64 grids. According to the distribution of the geocoding grids, the density map can be obtained. Each true location is obfuscated by the two DPG schemes. The true location is first encoded with geocoding system into a code c and locally differentially private noise with privacy budget ϵ is injected to the code c. In gray DPG, we set the length g of the suffix as 4. In other words, we perturb the original location data with different values of bit flipping probability $\frac{1}{e^{\frac{\epsilon}{4}}+1}$. In Geohash DPG, we set the hidden range h as 16. The comparison of worker density maps with different privacy level ϵ values are shown in the figure. Generally, we can find that when the privacy level ϵ is higher, the density map is more similar to that generated by workers' true location data. To give an intuitive view, we quantify the performance of the two DPG schemes with mean square error (MSE) between the obfuscated worker density map and the true worker density map. As shown in Figure 16, with higher ϵ value, the MSE is smaller and the performance of Geohash DPG is better than gray DPG.

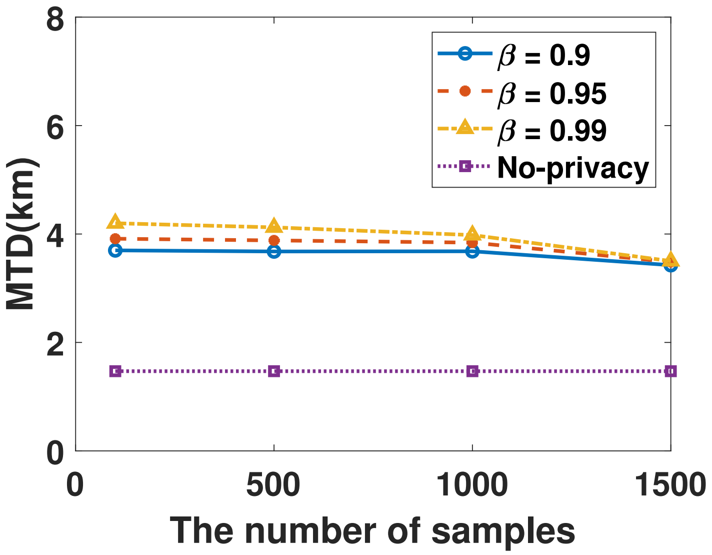


Figure 18: Impact of the β values under different number of samples.

In our work, we formulate a travel distance minimization problem. Because of the uncertainty brought by the location privacy preservation schemes, it is difficult to solve the optimization problem. Therefore, we leverage CVaR to characterize the uncertainty. Figure 17 demonstrates the performance comparison on the mean travel distance (MTD) under different parameters when applying the proposed two DPG schemes to protect workers' location privacy. In the simulations for Figure 17, when we are discussing the impact of one parameter, we control the other parameters with default values. Recall that in the default setting, we have 100 workers, 10 task assignments, confidence level β as 0.9, the number of samples in CVaR as 500 and the privacy level ϵ as 5. Figure 17a depicts the MTD of two proposed DPG schemes and the No-privacy scheme against the number of samples when approximating CVaR. As discussed in Section 6.4.5, CVaR is approximated by the equation (31). We can find that with more samples, the MTD of the two DPG schemes decreases and closer to that of No-privacy scheme, because with more samples the approximated value is supposed to approach the true CVaR value. Figure 17b illustrates the MTD of two proposed DPG schemes and the No-privacy scheme against the confidence level β. It shows that when confidence value increases, the MTD of the two DPG schemes increases since higher confidence level means that the confidence intervals would include the

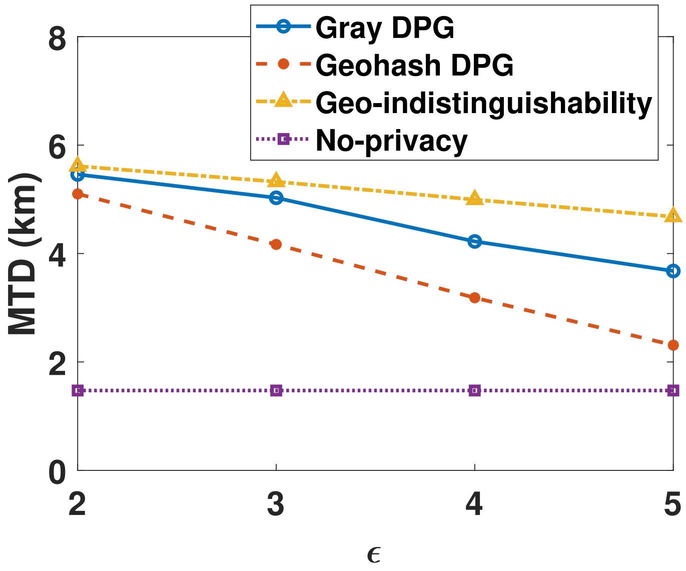


Figure 19: Impact of differential privacy level ϵ.

true travel distance with higher probability. Figure 17c indicates the MTD of two proposed DPG schemes and the No-privacy scheme against the number of task assignments. We can see that the trend of the three schemes are similar and the correlation between the number of tasks and MTD is not very high, as other elements may influence the results such as the distribution of task locations. Overall, Figure 17 shows that the MTD of the two proposed DPG schemes is always larger than that of No-privacy scheme and the Geohash DPG has a lower MTD than gray code DPG. The location privacy of workers is protected while sacrificing the task allocation efficiency. In Figure 18, we present the influence of the confidence level under different number of samples on the MTD with the gray code DPG. It reveals that with more sample to approximate CVaR the gaps between different confidence levels are shrinking. According to Figure 17 and 18, we can observe that in order to achieve a lower MTD, we can have more samples to approximate the CVaR and downsize the confidence level. Figure 19 plots the MTD of the two proposed DPG schemes and the other two benchmark schemes against different values of ϵ. We can find that the two proposed DPG schemes outperforms Geo-I scheme. With loosening the privacy budget ϵ, the MTD decreases. Moreover, when $\epsilon = 5$, the MTD of Geohash DPG is much smaller than the other two schemes and very close to the MTD of No-privacy scheme.

6.6 Conclusion

In this work, we studied an optimal MCS task allocation problem by minimizing workers' travel distance without compromising workers' location privacy. Hence, a novel approach called differentially private geocoding scheme (DPG) is proposed to protect workers' location privacy regardless of adversaries' prior knowledge, without relying on any trustful third-party. We develop two DPG schemes based on different geocoding systems. Moreover, we use conditional value at risk (CVaR) to mitigate the negative effect induced by DPG. We conduct comprehensive rigorous analysis based on the real-world dataset for taxi cabs. The impact of specific parameters in CVaR is discussed. The evaluation results verify the effectiveness of the two proposed DPG mechanisms.

7 Data-Driven Vehicle Scheduling with Users' Location Privacy

7.1 Introduction

The last decade has witnessed the exploding growth in the quantity and capability of consumer mobile devices such as smartphones, tablets, etc., and the proliferation of wireless services. With the advance and commercial use of global positioning system (GPS) technology, smartphones and tablets feature sensors that can pinpoint users' locations, which can allow the transportation network company (TNC) services to use users' whereabouts for ridesharing. Such TNC services (e.g., Uber, Lyft, Didi, etc.) can pair TNC users and TNC vehicles according to their location information through the mobile apps in order to provide ridesharing [12]. With TNC services, the users are able to save time by changing parking to TNC user loading events. As TNC users can gain more time savings when using TNC services, it also provide more job opportunities for drivers. In [13], the authors note that ridesharing scheduled by TNC can occupy the capacity more efficiently than traditional taxicabs. In addition, due to the development of technologies of self-driving cars, Google autonomous driving projects Waymo has already been testing in Arizona. The reported status shows that TNC is supposed to launch the self-driving car services sooner or later. Therefore, it is necessary to have a new TNC vehicle scheduling scheme for both the ridesharing drivers and self-driving vehicles with high efficiency and flexibility.

The explosive growth of usage of ridesharing services has created a new vehicle scheduling problem for TNC with high service guarantee. The TNC users' demands in different areas are supposed to change due to time, weather, events and etc. It is challenging to schedule TNC vehicles under the uncertainty of users' demand. In addition, the purpose of TNC is providing good service quality, hence vehicle supply should satisfy the users' demands in a specific area. There is always an assumption that the uncertain users' demands follow a carefully selected distribution that can fit the historical data in traditional stochastic optimization problem. However, the selected distribution cannot represent the true demands distribution owing to the limited amount of historical data.

Although users can gain benefits from TNC services, they also raise serious privacy concerns. Currently, most TNC services require the user's hand-held device to get the user's location from GPS, and periodically report the location information to the service provider, and the location data will be stored in the database/servers of the TNC service provider. Following this mechanism, a dishonest third-party service provider may have chances to leverage the user's reported locations and analyze the rich trace files of the user. With the exposure of locations, the users not only lose their privacy but also are vulnerable to various attacks, even some serious physical attacks [98]. For instance, if a celebrity registers for some dishonest services, he/she can be easily tracked by the paparazzi in the digital world. Correspondingly, his/her next location can be inferred, so that his/her privacy in the real world will be invaded. Another example is that, in August 2018, a violent robbery happened in Maryland. The victim is picked up by a fake Uber driver and the driver demanded the victim's wallet and cellphone with a handgun. In other words, the fake Uber driver is able to access to the users' location data. Besides robbery, it is not hard to imagine that there might be more serious crimes such as sexual assault, kidnapping, murder, assassination, etc. targeting specific victims at selected locations, due to the disclosure of users' location information. Additionally, as the TNC aims to pair TNC users and TNC vehicles efficiently and effectively, it also requires the high service quality. However, it is always challenging to have good service quality, while preserving user's privacy [111, 120].

To avoid those issues, it is worthy to protect the location data from the dishonest service providers or eavesdroppers. To thwart the dishonest service provider or eavesdropper, we exploit the geo-indistinguishability scheme [111] to add differential noise on users' location, which satisfies the "paradoxical" requirements of TNC users with location privacy. Moreover, we need to develop a new TNC vehicle scheduling scheme to satisfy users by reducing the waiting time. Therefore, it is necessary to innovate a TNC vehicle scheduling scheme to preserve users' location privacy without compromising the quality of services. However, with the geo-indistinguishability location privacy preservation method, users' location is perturbed by differential noise, hence there comes a challenge that the demand distribution uncertainty stems from two parts, the distance between the real demand distribution and the constructed reference distribution, and the differential noises introduced by location privacy preservation scheme. Therefore, we leverage three distance measurement metrics, Kantorovich metric, Fortet-Mourier metric and uniform metric to characterize the uncertainty. Aiming to address those challenges, we integrate geo-indistinguishability scheme and risk-averse two-stage data-driven optimization approach [121] to provide TNC vehicle scheduling with consideration of quantitatively analyzing the service quality, maximizing the TNC revenue under the uncertainty of users' demand, and preserving TNC users' location privacy by differential privacy settings. Our salient contributions are summarized as follows:

- We integrate geo-indistinguishability scheme based on differential privacy with data-driven optimization approach in the TNC service scenario. With geo-indistinguishability, on the TNC users' side, the TNC users' exact locations can be hidden in a range with radius r according to the privacy level. Additionally, on the TNC's side, data-driven optimization method is deployed to efficiently and effectively schedule the TNC vehicles' cruising plan. Therefore, a total revenue maximization problem is formulated with the consideration of reducing TNC users' waiting time, which means the quality of TNC services can be guaranteed. In this work, we aim to take better advantage of the TNC users' obfuscated location data, and meanwhile the location privacy can be protected with the concept of differential privacy.
- Because of the limited amount of historical data and the noise generated by location privacy preservation scheme, the users' demand distribution is uncertain. Therefore, risk-averse two-stage data-driven approach is deployed to characterize the data uncertainty. We assume the TNC constructs the ridesharing demand distribution $\mathbb{P}_0$ of a location range, which is the dashed yellow curve (the private reference distribution) shown in Figure 20, according to the obfuscated TNC users' location information. The ambiguous distribution $\mathbb{P}$ is the blue curve in Figure 20, which is unknown. The

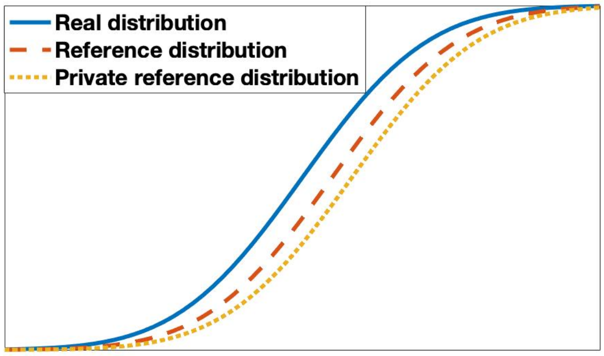


Figure 20: Reference distribution.

uncertainty of the demand is characterized by distance between the ambiguous distribution $\mathbb{P}$ and the private reference demand probability $\mathbb{P}_0$. Consequently, we can solve the risk-averse two-stage revenue maximization problem. In the first stage, TNC schedules the vehicles to go to a particular location range. In the second stage, in order to reduce the waiting time of each user and guarantee the QoS, when the true demand of an area is higher, other vehicles will be sent to the particular area.

- The formulated revenue maximization problem can be converted into a risk-averse two-stage stochastic problem (RA-SP). In other words, we maximize the total revenue in a worst-case condition, which is different from other distribution-free scheduling approaches. The the proposed problem is solved with three different distribution distance metrics under ζ-structure for robustness.

- We conduct simulations with the real public data from Didi ridesharing company to verify the effectiveness of the proposed scheme.

The rest of work is organized as follows. We review the related work on location privacy and TNC services in Section 7.2. In Section 7.3, we present the overview of our system and describe the preliminaries. In Section 7.4, we give the formulation of the revenue maximization problem, apply data-driven methodology to feasibly solve the problem. In Section 7.5, we analyze the performance evaluation. Finally, we draw conclusions in Section 7.6.

Table 4: Notation List

Symbol	Definition
$\mathcal{K}$	Set of TNC vehicles
c_k	Cost for a driver
X	The number of other vehicles sent to fulfill the demand
q_x	Cost for each available vehicle sent later
$D(\xi)$	Private demand distribution of a specific area
$\mathbb{P}_0$	Reference distribution of demand
$\mathbb{P}$	Ambiguous true distribution of demand
ϵ	DP privacy budget
$\mathcal{D}$	Confidence set
η	Confidence level
d_ζ	Distribution distance under ζ-structure probability metric
θ	Convergence rate
Ω	The sample space of ξ
$\varnothing$	The dimension of Ω

7.2 Related Work

As the ridesharing services provide significant benefits to the transportation system and environment and the technology of self-driving vehicles become mature, TNC vehicle scheduling gained tremendous popularity [122, 123, 124]. In [123], the authors proposed a large-scale taxi scheduling and formulated a dynamic ridesharing problem in order to reduce the total cruising distance of drivers. Therefore, the satisfactory ratio can be guaranteed. Moreover, as the self-driving technology becomes mature, a new public transportation system based on autonomous vehicles is proposed in [124]. The aim of the work is to decide the most economical schedules for self-driving cars in order to maximize the total revenue. This admission and scheduling control problem is formulated into a mixed-integer linear program and the authors give a feasible solutions with the real Boston Taxi data. While the TNC users are using the ridesharing services, their private location information is disclosed. In [98], the authors deploy differential privacy to protect users' location privacy while minimizing the total cruising distance of TNC drivers during the ridesharing scheduling. Compared to this work, we give a more robust solution to our formulated ridesharing revenue maximization problem under the worst-case with a limited historical data set.

With the proliferation of ridesharing services and location-based services (LBS), many works focus on location privacy preservation schemes recently, where the TNC and LBS provider are considered to be honest-but-curious. The location protection approaches can be divided into three categories. The first one is obfuscation-based method, which is also called dummy-based method [125, 100]. In this case, users send dummy requests together with the true request, and and hence the attacker cannot distinguish the real location from the dummy locations. In [11], the authors obfuscated the workers' location privacy in mobile crowdsensing application under the differential private setting, while minimizing the workers' total travel distance. However, this kind of approaches will compromise the quality of service because of the inaccuracy of the location data. Moreover, in [126], the authors proposed an attack model which can bound a users in a specific area despite the obfuscation with dummy queries. The second category is collaboration-based methods, where each user sends a time or space obscure cloak region to the servers instead of the true location [102, 103, 104]. In [102], the authors proposed a user-collaborative privacy-preserving approach that LBS users can seek information directly from their nearby peers and when the users cannot obtain the information from peers, they would query the LBS. The authors in [127] proposed a SafeBox scheme based on spario-temporal generalization to protect user's location in a nonsensitive geographical area and a time interval, while keep the utility of LBS. Generally speaking, in collaboration-based methods, it may need additional high cost preprocessing of the data which may further incur high communication cost. The last kind of methods is identity and location anonymity. The mix-zone model [105, 106] is first proposed to be used in location privacy preservation in [107]. A mix-zone indicates that when users enter the mix-zone, they can change their pseudonym to prevent adversary from tracking their locations. Furthermore, some schemes belonging to this category often put the true location together with another $k-1$ dummy locations in an area from historical data or other users' location to guarantee k-anonymity [108]. The main concern in this category is that a trustworthy anonymizer is required to construct the anonymity settings, nevertheless the anonymizer can also be the malicious party. To address this concern, differential privacy is leveraged in the location privacy preservation schemes, which have been studied in [111, 128]. In our work, since we protect users' location privacy with differential privacy mechanism, a trustworthy third party is not required in our system. Moreover, we utilize data-driven optimization to characterize the uncertainty brought by the differential private noise, hence the quality of service can be guaranteed.

7.3 Network Model and Preliminaries

In this section, we describe our network model for TNC vehicle scheduling and then introduce the differential privacy based geo-indistinguishability scheme.

7.3.1 System Description

The purpose of TNC services is to pair the users and vehicles including ridesharing and self-driving vehicles via mobile apps. It is supposed to provide ride-sourcing services that can serve the users best, which means that the provided service can reduce TNC users' waiting time and schedule the TNC vehicles' cruising plan effectively and efficiently. In our work, we propose an efficient TNC vehicle scheduling scheme and formulate the revenue maximization problem. Moreover, in order to protect the TNC users' location privacy, a differential privacy based location privacy preservation scheme called geo-indistinguishability is exploited, which will be introduced in Section 7.3.2. With the obfuscated location data, the demand of hailing vehicles in an area can be predicted. As the TNC users' location is obfuscated by the scheme, the demand of vehicles in a certain location range is uncertain. Therefore, the data-driven methodology is employed to characterize the demand uncertainty.

In our architecture, the TNC users report their obfuscated location to the TNC server. In order to protect TNC users' location privacy, the geo-indistinguishability is applied to obfuscate a TNC user's true location. There are several vehicles cruising nearby and the

TNC users can hail a vehicle through the TNC mobile apps or websites. The TNC server will construct a reference demand distribution $\mathbb{P}_0$ based on the users' obfuscated location. According to the predicted demand, several TNC vehicles can be scheduled to a specific region. As the true demand of each region is uncertain, the scheduled vehicles may not meet the demand of all TNC users. In this kind of situation, other vehicles is required to be sent to the area so as to meet the current demand. Therefore, users' wait time can be reduced and the QoS is assured.

7.3.2 Geo-indistinguishability Preliminaries

Geo-indistinguishability scheme [111, 112] is defined based on the differential privacy concept, which can protect location privacy in despite of side information known by the attackers. The main purpose of the scheme is to obfuscate the user's location in an area with radius $r \geq 0$, and hence ϵr-location privacy can be guaranteed applying the scheme. As ϵ is the privacy budget of differential privacy that represents the privacy level, the location privacy level can be defined as l which means that in user's concerned location range within the radius r, l privacy level is needed to meet the user's privacy satisfactory. In other words, within range r, the geo-indistinguishability scheme can guarantee l-privacy with $l = \epsilon r$. We assume that the attackers have the side information, which can be defined as the prior distribution π on the users' frequent location set $\mathcal{X}$. The randomized mechanism satisfies $\mathcal{A}(x) \in \mathcal{Z}$, where $\mathcal{Z}$ is the private location set. We represent the probability that the randomized output of x in the location set $\mathcal{Z}$ as $\mathcal{A}(x)(\mathcal{Z})$. The posterior distribution can be calculated by $\sigma(x) = \text{Bayes}(\pi, \mathcal{A}, \mathcal{Z}) = \frac{\mathcal{A}(x)(\mathcal{Z})\pi(x)}{\sum_{x' \in \mathcal{X}} \mathcal{A}(x')(\mathcal{Z})\pi(x')}$. The multiplicative distance between two distribution on set $\mathcal{S}$ is denoted as $d_m(\sigma_1, \sigma_2) = \sup_{S \subseteq \mathcal{S}} |\ln \sigma_1(S)/\sigma_2(S)|$. Intuitively, the definition of geo-indistinguishability can be modified from Definition 2.1, which can be described as follows.

Definition 7.1 (Geo-indistinguishability [111]) With a privacy confidence parameter $\epsilon \geq 0$, a randomized algorithm $\mathcal{A}$ satisfies ϵ-geo-indistinguishability, when given two different locations x and x', in the following condition,

$$\begin{aligned}
&d_m(\mathcal{A}(x), \mathcal{A}(x')) \leq \epsilon d(x, x'), \\
&\text{s.t.,} \\
&\mathcal{A}(x) \in Z, \mathcal{A}(x') \in Z,
\end{aligned}$$

where $d(\cdot, \cdot)$ is the Euclidean distance.

To achieve l-privacy within an area with radius r, given two different locations x and x', the outputs from the randomized algorithm should be indistinguishable. The definition forces that the distance between two locations x and x' is not larger than r.

7.3.3 Threat Model

During the TNC scheduling, the server will pair users with vehicles. The server can obtain users location information continuously when they are using the services. In addition, some apps run in the background may still upload users' location information to the server. These circumstances could cause privacy disclosure. We assume that the attacker could be an outside entity or the dishonest TNC server. In other words, the adversaries have the access to the users' location data and have the ability to infer other information according to the true location data. We consider that the adversary may have users' arbitrary side information. Therefore, in our work, we focus on protecting users' location privacy before they are paired with vehicles. To prevent information leakage after pairing is out of our scope, because a user and a paired vehicle will reveal information to each other. We also assume there is a secure communication channel between a user and a specific paired vehicle.

7.4 Data-Driven TNC Revenue Maximization Optimization Problem with Differential Privacy

In Section 7.3.2, the main purpose and definition are introduced. In this section, we formulate the revenue maximization problem with differential private TNC users' location data in a RA-SP model. We give a feasible solution to the problem by exploiting data-driven methodology.

7.4.1 Protecting Location Privacy with Differential Privacy

In general, in order to achieve this l-privacy guarantee, the probability of obfuscated location of true location x_0 should exponentially decrease along the radius r. In other words, the probability of generated the obfuscated location close to true location x_0 is higher. According to the Laplace distribution with the probability density function (pdf), the probability to generate the obfuscated location distributes is $\frac{\epsilon}{2}e^{-\epsilon|x-\mu|}$. In our scenario, the Laplace noise is added on location data. Therefore, a two-dimensional Laplace distribution is supposed to be employed. The pdf of two-dimensional Laplace distribution can be derived from the linear one by replacing $|x-\mu|$ with the distance $d(x, x_0)$, which is shown as

$$D_\epsilon(x_0)(x) = \frac{\epsilon^2}{2\pi} re^{-\epsilon d(x_0,x)}. \tag{38}$$

It is obvious that the distance between x_0 and x is the only parameter in the pdf of high dimensional Laplace distribution. Therefore, converting the Cartesian coordinates to polar coordinates with original at x_0 should be convenient and intuitive [111]. The pdf of high dimensional Laplace distribution in polar coordinate can be represented as

$$D_\epsilon(r,\theta) = \frac{\epsilon^2}{2\pi} re^{-r\epsilon}. \tag{39}$$

The pdf clearly show that the two random variables r and θ are independent. Therefore, the pdf can be decomposed into two parts with the two random variables. The representations are shown as

$$D_{\epsilon,R}(r) = \int_0^{2\pi} D_\epsilon(r,\theta)d\theta = \epsilon^2 re^{-r\epsilon} \tag{40}$$

and

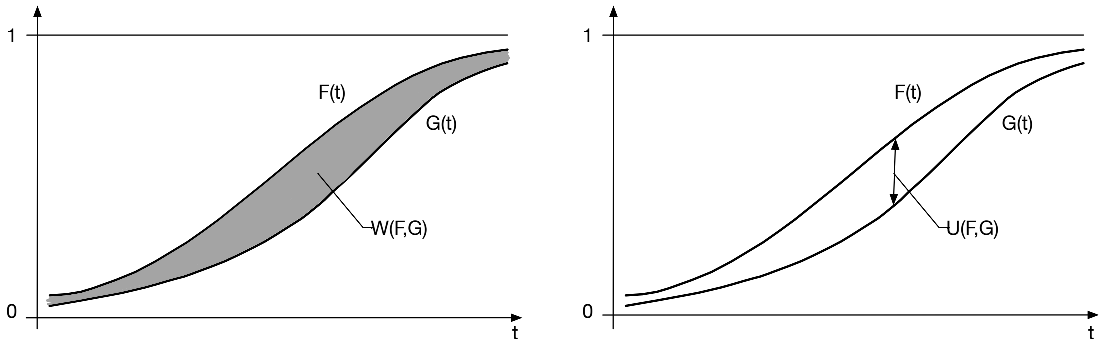
$$D_{\epsilon,\Theta}(\theta) = \int_0^\infty D_\epsilon(r,\theta)dr = 1/2\pi. \tag{41}$$

Due to the independence of the two random variables, the Laplace noise can be generated separately with r and θ. The distribution of Θ is a constant, and hence the noise added to angle can be uniformly choose from the interval $[0, 2\pi)$. In order to generate the noise added on the distance, we first calculate the cumulative distribution function (cdf) of r shown as

$$C_\epsilon(r) = \int_0^r D_{\epsilon,R}(\rho)d\rho = 1 - (1 + r\epsilon)e^{-r\epsilon}. \tag{42}$$

This cdf denotes the probability of the random point drawn between 0 and r. Here, r is supposed to be $C_\epsilon^{-1}(p)$, where p is uniformly selected from the interval $[0, 1)$. Here, we have $C_\epsilon^{-1}(p) = -\frac{1}{\epsilon}(W_{-1}(\frac{p-1}{e}) + 1)$, where W_{-1} is the Lambert W function with -1 branch. In this case, the noise (r, θ) can be randomly selected with the high dimensional Laplace distribution. Therefore, we can get the obfuscated location z from the original location x_0 by adding noise (r, θ) from polar coordinate, which can be represented as $z = x_0 + < r\cos\theta, r\sin\theta >$.

Due to the finite precision of the machine, when generating the Laplace noise, it is discretized. Therefore, we need to provide noise compensations for the influences of this discretization. In [111], the authors showed that with double precision, the noise compensation is negligible. To satisfy ϵ-geo-indistinguishability with the discretized mechanism, [111] provided truncation mechanism to generate obfuscated location within a specified finite region. If the obfuscated location is outside this region, it will be remapped to the closest point in the predefined specified finite region. This is the full mechanism with discretization and truncation, which satisfies the definition of ϵ-geo-indistinguishability.



(a) Wasserstein metrics (one-dimensional case).

(b) Uniform metric (one-dimensional case).

Figure 21: Comparison of two metrics.

7.4.2 Data-driven Analysis of Demand Prediction

In practice, the number of rider demand in a location at a specific time is unknown and with uncertainty. In order to guarantee the quality of TNC services and maintain the scheduling scheme reliability, in our work, we employ data-driven risk-averse stochastic program (RA-SP) approach to making a decision to schedule the vehicles under the uncertainty of predicting the demand. In other words, we consider the worst-case probability distribution of the demand and our solution to this TNC vehicle scheduling problem is feasible and robust.

As it is hard to get the true probability distribution of a real-life random variable because of the limited available historical data, in [129], the authors propose the RA-SP optimization approach to characterize the distribution ambiguity. A confidence set $\mathcal{D}$ is used to describe that with a certain confidence level the true probability distribution is supposed to be in the confidence set $\mathcal{D}$. The confidence set can be constructed by the distance measure between two distributions. The distance between two distributions under ζ-structure can be represented as $d_\zeta(\mathbb{P}_0, \mathbb{P})$, where $\mathbb{P}_0$ is the private reference distribution generated from the historical data and $\mathbb{P}$ is the ambiguous distribution of the demand of hailing vehicles in a specific area. The confidence set $\mathcal{D}$ and distance measure under the

ζ-structure probability metrics can be defined as

$$\mathcal{D} = \{\mathbb{P}_i : d_\zeta(\mathbb{P}_0, \mathbb{P}) \leq \theta\} \tag{43}$$

and

$$d_\zeta(\mathbb{P}_0, \mathbb{P}) = \sup_{h \in \mathcal{H}} \left| \int_\Omega h d\mathbb{P}_0 - \int_\Omega h d\mathbb{P} \right|, \tag{44}$$

where θ is the tolerance level which is decided by the amount of historical data, and $\mathcal{H}$ is a real-valued bounded measurable functions family on Ω (the sample space on ζ). Therefore, with higher amount of historical data, the confidence set $\mathcal{D}$ should be tighter and the distance between $\mathbb{P}_0$ and $\mathbb{P}$ should be smaller.

As we illustrated before, $\mathcal{H}$ is a real-valued bounded measurable functions family. We employ three different metrics, Kantorovich metric, Fortet-Mourier metric and uniform metric, in the family $\mathcal{H}$ to solve the proposed optimization problem. For a random variable x, the distribution can be represented as $\mathbb{P} = \mathcal{L}(x)$. The distance between two random variables x and y is set as $\rho(x, y)$.

- **Kantorovich metric** is first derived to relax the formulation of the *transportation problem* and solve the problem to find the optimal allocation of resources [130]. The distribution distance is represented as $d_K(\mathbb{P}_0, \mathbb{P})$, $\mathcal{H} = \{h : ||h||_L \leq 1\}$, where $||h||_L := \sup\{h(x) - h(y)/\rho(x, y) : x \neq y \text{ in } \Omega\}$. When $\Omega = R$, the Wasserstein metric is the same as Kantorovich metric because of the Kantorovich-Rubinstein thoreo. The distribution distance under Wasserstin metric can be denoted as $d_w(\mathbb{P}_0, \mathbb{P}) = \int_{-\infty}^{+\infty} |F(x) - G(x)|dx$, where F and G are the distribution function derived from $\mathbb{P}_0$ and $\mathbb{P}$ respectively, which is demonstrated in Figure 21a.

- In [131], the authors exploited **Fortet-Mourier metric** to reduce the scenarios when solving a convex stochastic programming problem. The distribution distance measure of this metric is defined as $d_{FM}(\mathbb{P}_0, \mathbb{P})$, $\mathcal{H} = \{h : ||h||_C \leq 1\}$, where $||h||_C :=$

$\sup\{h(x) - h(y)/c(x,y) : x \neq y \text{ in } \Omega\}$ and $c(x,y) = \rho(x,y)\max\{1, \rho(x,a)^{p-1}, \rho(y,a)^{p-1}\}$ for $p \geq 1$ and $a \in \Omega$. Kantorovich metric is a special case of Fortet-Mourier metric when $p = 1$. Therefore, it is also widely employed on the *transportation problem.* The relationship between the two metrics [121] is represented as

$$d_{FM}(\mathbb{P}, \mathbb{Q}) \leq \Lambda \cdot d_K(\mathbb{P}, \mathbb{Q}), \tag{45}$$

where $\Lambda = \max\{1, \varnothing^{p-1}\}$ and $\varnothing$ is the diameter of Ω.

- In **Uniform metric**, the distance measure is defined as $d_U(\mathbb{P}_0, \mathbb{P})$, $\mathcal{H} = \{I_{(-\infty,t]}, t \in R^n\}$, which is shown in Figure 21b. According to the definition, we have $d_U(\mathbb{P}_0, \mathbb{P}) = \sup_t |\mathbb{P}_0(x \leq t), \mathbb{P}(x \leq t)|$.

According to the description of each metrics under ζ-structure, we need to figure out the value of tolerance level θ, which can also be used to quantify the convergence rate. For the uniform metric, the convergence rate can be derived from the Dvoretzky-Kiefer-Wolfowitz inequality [132], which is shown as (i.e., $n = 1$),

$$P(d_U(\mathbb{P}_0, \mathbb{P}) \leq \theta) \geq 1 - 2e^{-2Q\theta^2}, \tag{46}$$

where Q is the number of historical data.

In [121], the converge rate of the Kantorovich metric for a general dimension case (i.e., $n \geq 1$) is defined as

$$P(d_K(\mathbb{P}_0, \mathbb{P}) \leq \theta) \geq 1 - \exp\Big(-\frac{\theta^2 Q}{2\varnothing^2}\Big). \tag{47}$$

Consequently, due to the relationships between the Kantorovich metric and the Fortet-Mourier metric, the convergence rates of the two metrics can be calculated accordingly.

Corollary 1 For a general dimension (i.e., $n \geq 1$), we have

$$P(d_{FM}(\mathbb{P}_i^0, \mathbb{P}_i) \leq \theta) \geq 1 - \exp\Big(-\frac{\theta^2 Q}{2\varnothing^2\Lambda^2}\Big). \tag{48}$$

With the knowledge of distribution distance measure and the convergence rates, the value of tolerance level θ can be calculated. We have the confidence level is set to η that is represented as $\mathbb{P}(d_K(\mathbb{P}_i^0, \mathbb{P}_i) \leq \theta) \geq 1 - \exp(-\frac{\theta^2}{2\varnothing^2}Q) = \eta$. The value of θ can be obtained as $\theta = \varnothing\sqrt{2log(1/(1-\eta))/Q}$.

7.4.3 TNC Revenue Maximization Problem with Location Privacy Preservation

So far, the geo-indistinguishability scheme is introduced and the data-driven RA-SP approach is described. In this subsection, we will give the formulation of TNC revenue maximization problem with location privacy preservation. As we describe before, the TNC users' location data is obfuscated by the geo-indistinguishability scheme. The map is divided into several small regions r, which in a set $\mathcal{R} = \{1, \cdots, r, \cdots, R\}$. With the obscure location data, the demand of hailing vehicles in a specific area r can be predicted and the private reference demand distribution can be constructed as $\mathbb{P}_0^r$. As the demand map is generated from TNC users' obfuscated location data, data-driven methodology is employed to solve the uncertainty problem. Moreover, the demand of hailing vehicles corresponding to the location data can be represented as $D_r(\xi)$. The TNC server will make decisions to send y_r vehicles to go to the particular area according to the average cost for sending vehicles to the specific region which is represented by c_r. Here, the average cost c_r is based on the average fare f_r of each trip. As the private reference demand distribution is constructed by the obfuscated location data, data-driven methodology is applied to characterize the uncertainty of the distribution. In order to provide better service to the TNC users, the waiting time of each user should be reduced. Therefore, when the TNC server observes a high-demand scenario such as the morning rush hour, the true demand of such a region is high and y_r vehicles are not enough for this situation, hence the TNC server decides to send another x_r vehicles to satisfy the demand with a surge cost βc_r. The revenue maximization problem for TNC can be formulated as follows,

$$\max_{y}\sum_{r=1}^{R}-c_r y_r+\min_{\mathbb{P}\in\mathcal{D}}\mathbb{E}_{\mathbb{P}}\max_{x}[f_r\min(x_r+y_r,D_r(\xi))-\beta c_r x_r], \tag{49}$$

s.t.,

$$x_r\geq 0, y_r\geq 0, \forall r. \tag{50}$$

In the formulation, y_r and x_r are integer variables, which are supposed to be positive numbers. Since the TNC users' injected noises to their location data, the distribution of real demand is ambiguous. Therefore, we construct the confident set $\mathcal{D}$ with $\zeta-$struction probability metrics introduced in 7.4.2, and let $\mathbb{P}\in\mathcal{D}$ so as to maximize the total revenue under the worst-case distribution realization in $\mathcal{D}$.

7.4.4 Solution to the Proposed Optimization Problem

In order to solve the proposed revenue maximization problem, assuming the sample space with N scenarios as $\Omega=\{\xi^1,\xi^2,\cdots,\xi^N\}$, we reformulate the problem according to the ζ-structure metrics shown as follows,

$$\max_{y}\sum_{r=1}^{R}-c_r y_r+\min_{p_i}\sum_{i=1}^{N}p_i\max_{x}[f_r\min(x_r+y_r,D_r(\xi^i))-\beta c_r x_r], \tag{51}$$

s.t.: (50),

$$\sum_{n=1}^{N}p_i=1, \tag{52}$$

$$\max_{h_i}\sum_{i=1}^{N}h_i p_i^0-\sum_{i=1}^{N}h_i p_i\leq\theta, \forall h_i:||h||_\zeta\leq 1, \tag{53}$$

where θ is the tolerance level. According to definition of different ζ-structure metrics, $|h||_\zeta$ is defined according to different metrics. For the Kantorovich metric, $|h_i-h_j|\leq\rho(\xi^i,\xi^j)$. For the Fortet-Mourier metric, $|h_i-h_j|\leq\rho(\xi^i,\xi^j)\max\{1,\rho(\xi^i,a)^{p-1},\rho(\xi^j,a)^{p-1}\}$. The constraints (53) can be summarized as $\sum_{i=1}^{N}a_{ij}h_i\leq b_j, j=1,\cdots,J$. To reformulate the constraints, we consider the following problem,

$$\max_{h_i} \quad \sum_{i=1}^{N} h_i p_i^0 - \sum_{i=1}^{N} h_i p_i, \tag{54}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} a_{ij} h_i \leq b_j, j = 1, \cdots, J. \tag{55}$$

The dual problem can be formulated with a dual variable u, which is shown as follows,

$$\min_{u} \quad \sum_{j=1}^{J} b_j u_j, \tag{56}$$

$$\text{s.t.} \quad \sum_{j=1}^{J} a_{ij} u_j \geq p_i^0 - p_i, \forall i = 1, \cdots, N. \tag{57}$$

Accordingly, the problem can be reformulated as follows under ζ-structure metrics Kantorovich metric and Fortet-Mourier metric,

$$\max_{y} \sum_{r=1}^{R} -c_r y_r + \min_{p_i} \sum_{i=1}^{N} p_i \max_{x} [f_r \min(x_r + y_r, D_r(\xi^i)) - \beta c_r x_r], \tag{58}$$

$$\text{s.t.} \quad (50), (52)$$

$$\sum_{j=1}^{J} b_j u_j \leq \theta, \tag{59}$$

$$\sum_{j=1}^{J} a_{ij} u_j \geq p_i^0 - p_i, \forall i = 1, \cdots, N, \tag{60}$$

Based on the definition of Uniform metric, the problem can be reformulated as follows,

$$\max_{y} \sum_{r=1}^{R} -c_r y_r + \min_{p_i} \sum_{i=1}^{N} p_i \max_{x} [f_r \min(x_r + y_r, D_r(\xi^i)) - \beta c_r x_r], \tag{61}$$

$$\text{s.t.} \quad (50), (52)$$

$$\left| \sum_{i=1}^{j} \left(p_i^0 - p_i\right) \right| \leq \theta, \forall j. \tag{62}$$

Consequently, the proposed revenue maximization problem can be feasibly solved by the reformulations based on the three different ζ-structure metrics. The procedures to solve the proposed optimization problem can be summarized in Algorithm 5.

Algorithm 5 Solutions to the revenue maximization problem

```
Input: User's obfuscated historical location data, the confident level $\eta$.
Output: Objective value of the expected revenue.
Construct the private reference distribution $\mathbb{P}_0$ based on the noisy location data
Calculate the tolerance level $\theta$ with the confidence level $\eta$ under different $\zeta$-structure metrics
if Under Kantorovich metric or Fortet-Mourier metric then
    Solve the reformulated problem (58)
    Stop and return objective value
else
    Solve the reformulated problem (61) under Uniform metric
    Return the objective value.
end if
```

7.4.5 Security Analysis

In our work, we exploit the geo-indistinguishability to protect TNC users' location privacy. During the TNC users' location reporting phase, instead of uploading the true location, the obfuscated location is reported to the TNC server. Each user adds high dimensional Laplace noise to the true location. According to the geo-indistinguishability definition, the obfuscated location distribution is close to each other. In other words, with the same output obfuscated location, attackers cannot distinguish whether the input true location is x or x' even though the attackers have users' side information. We assume the attacker knows the users' frequent location set $N \subseteq \mathcal{X}$. In order to measure information leakage of the scheme, we can compare the prior and posterior distributions. The prior distribution of x in N can be represented as $\pi(x|N)$ and the posterior distribution can be represented by $\sigma(x|N)$. The maximum distance between any two locations in the set N is denoted as $d(N)$. Recall that the multiplicative distance between two distribution on set $\mathcal{S}$ is denoted as $d_m(\sigma_1, \sigma_2) = \sup_{S \subseteq \mathcal{S}} |\ln \sigma_1(S)/\sigma_2(S)|$. If a randomized mechanism $\mathcal{A}$ satisfies the ϵ-geo-indistinguishability definition, the following inequality is supposed to be

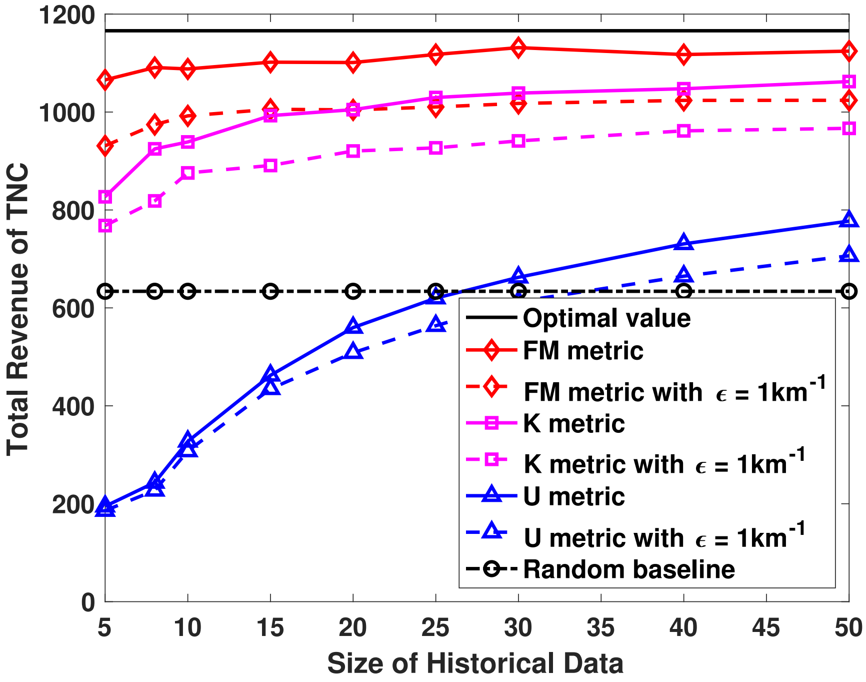


Figure 22: Total revenue in Chengdu under different metrics.

guaranteed,

$$d_m(\pi(x|N), \sigma(x|N)) \leq \epsilon d(N).$$

This inequality can be derived from the ϵ-indistinguishability definition, which is first informed in [15]. It illustrates that although the attacker knows all the other input data except x, he cannot obtain any information from x in spite of any side information of x. This inequality also represents that the adversary's obtained information is dependent on the range $d(N)$, in spite of the prior knowledge $\pi(x|N)$. Here, $d(N)$ represents the accuracy of adversary's side information. A small value of $d(N)$ means that the adversary has some accuracy of the users' location. When $d(N)$ is small, the distance between the prior and posterior distribution is also small, which means the adversary is hard to improve the accuracy of his knowledge. The performance of geo-indistinguishability will be discussed in Section 7.5.2.

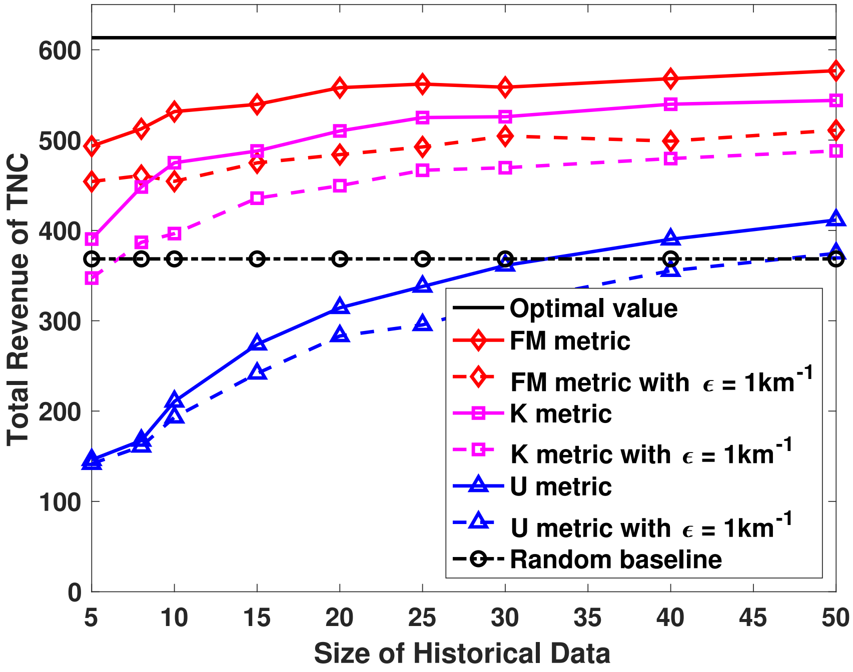


Figure 23: Total revenue in Xi'an under different metrics.

7.5 Performance Evaluation

In this section, we demonstrate our simulation results of the proposed scheme. We conduct the simulation by Matlab and GUROBI on a PC with Intel Core i7 CPU and 16.0 GB memory size.

7.5.1 Dataset and Simulation Setup

In the simulation, we implement our scheme with the public real location data from Didi transportation company during October and November 2016 in two cities, Chengdu and Xi'an [133]. The database contains each TNC driver's encrypted ID, the order ID, timestamp, longitude, and latitude. We investigate the proposed scheme in the downtown area in the two cities and during the morning rush hour from 8am to 9am. We randomly select 1000 users in the downtown area of each city. We divide the area into 5 × 5 grids and aggregate the demand of each region during the provided two months. To simplify the simulation, we select two grid of the two cities to conduct the experiment. We employ

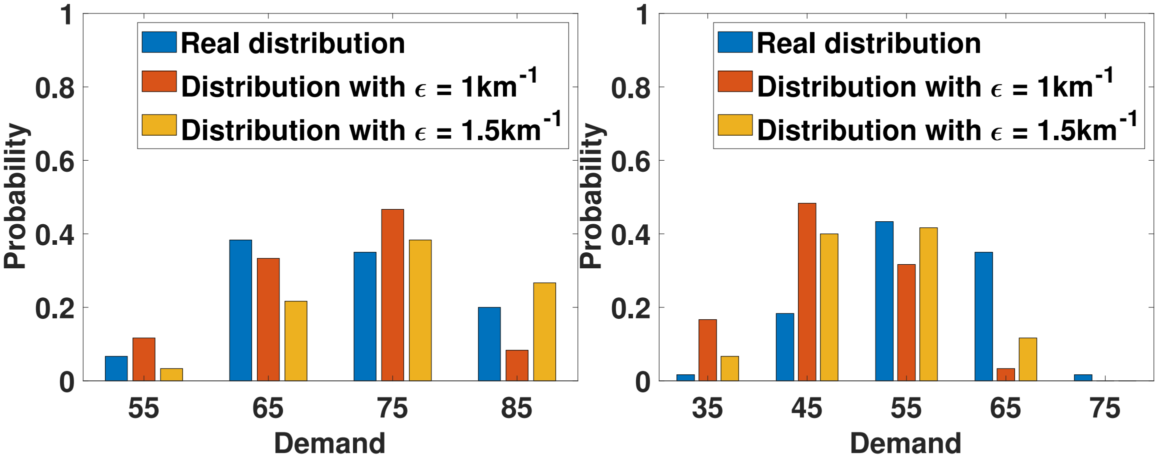


Figure 24: TNC vehicle demand distributions of two regions in Chengdu.

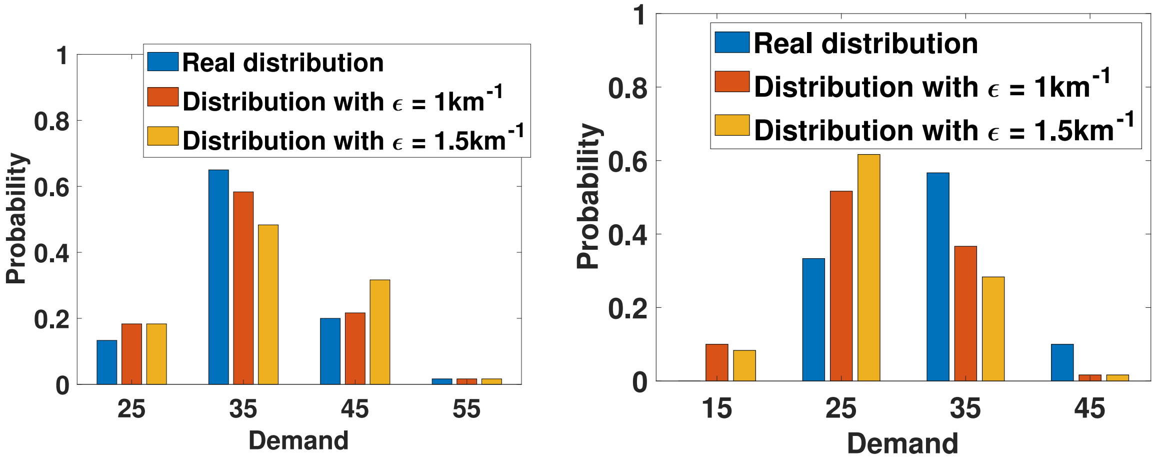
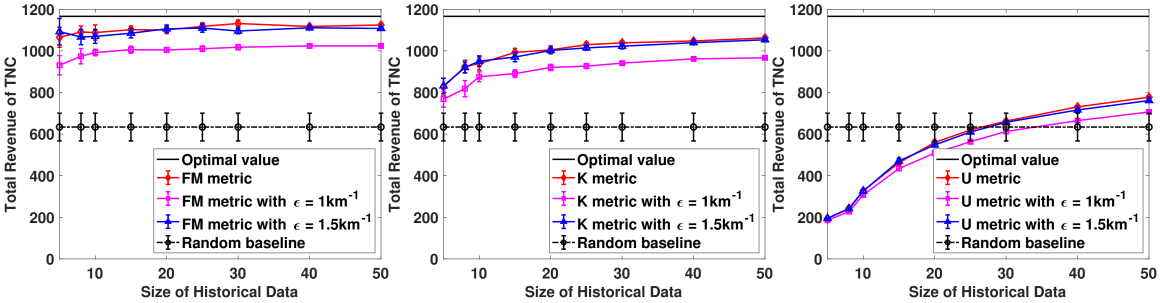


Figure 25: TNC vehicle demand distributions of two regions in Xi'an.

geo-indistinguishability introduced in Section 7.4.1 to protect users' location privacy. As demonstrated in [111], since we use double precision, the noise compensation can be neglected. Similar to the settings in [111], we set the radius of the specified finite region of the truncation mechanism as 100km. We consider the geo-indistinguishability scheme can protect users' location privacy within radius $r = 2.5$km. Then, we evaluate the privacy budget $\epsilon = [1, 1.5]\text{km}^{-1}$. We construct the confidence set $\mathcal{D}$ with confidence level $\eta = 0.9$ and three distribution distance measurement metrics, Kantorovich metric (K metic), Fortet-Mourier metric (FM metric) and uniform metric (U metric). We use the random scheduling as baseline for comparison in the simulation. In this scheduling method, we assume that the

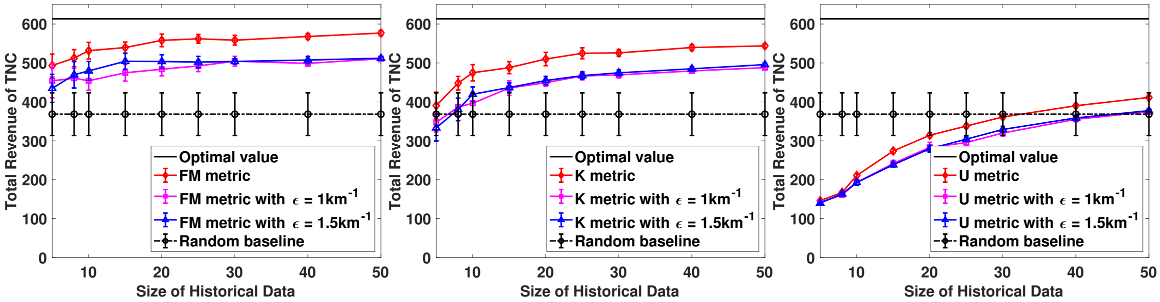


(a) Performance under the Fortet-Mourier metric.

(b) Performance under the Kantorovich metric.

(c) Performance under the Uniform metric.

Figure 26: Performance comparison in Chengdu of differential privacy under different metrics.



(a) Performance under the Fortet-Mourier metric.

(b) Performance under the Kantorovich metric.

(c) Performance under the Uniform metric.

Figure 27: Performance comparison in Xi'an of differential privacy under different metrics.

TNC server will schedule a random number of vehicles to a specific region based on the true demand distribution. Similar to our proposed revenue maximization problem, if this random number cannot satisfy the high demand scenario, the TNC server will send the required amount of vehicles later.

7.5.2 Performance evaluation

As described in Section 7.5.1, we investigate the TNC vehicle demand of the downtown areas of two cities. We randomly choose 1000 users of each city and we divide the downtown areas of the two cities into 5 × 5 grids. We add the high dimensional Laplace noise with differential privacy budget ϵ to the original location data. The private reference distribution $\mathbb{P}_0$ of TNC vehicle demand can be constructed according to the noisy location data. As described in Section 7.5.1, to simplify the simulation, we conduct simulations based on the demand distributions of the two chosen grids from each city. We conduct 50 independent runs of our scheme and show the average revenue values in Figure 22 and 23, which represent the performance of the expected total revenue under the three different metrics. We can observe that the obtained objective values returned by each metric are getting closer to the optimal results with more historical data, because with a larger amount of historical data, the tolerance rate θ is smaller, which makes the confident set tighter to the real demand distribution. In our simulation, we only have 60-day TNC order data, which is limited, and hence the results obtained by the ζ-structure metrics are not converged to the optimal result, especially with the U metric. In other words, with enough amount of historical data, the performance can show the convergence better and the TNC vehicle demand distribution can be more accurate. Moreover, the total revenue of TNC with FM metric is closest to the optimal values in both cities. We provide random scheduling as a baseline for comparisons. Since we assume that in random scheduling, a random number of vehicles will be scheduled to the small regions according to the true demand distribution, the total revenue of the random scheduling is not dependent on the historical data. Therefore, it is a straight line. If there is limited historical data, the random scheduling method may perform better than the uniform metric. When we have enough historical data, all the three metrics outperform the random scheduling.

Here, we investigate the performance of the two cities with different privacy budget ϵ. The TNC vehicle demand distributions in two cities are shown in Figure 24 and 25. The demand distributions are generated by the collected obfuscated location information. Then, we select two grids in each city to show the vehicle demand distributions. Generally, we can find that although there is a certain distance between the true demand distribution and the noisy demand distribution, the overall distributions are similar to each other. In Figure 24, with a higher ϵ value, which means lower privacy level, the demand distributions are much closer to the true demand distributions in Chengdu. The expectations of the two regions in Chengdu are approximately 129 without geo-indistinguishability, 117 with $\epsilon = 1\text{km}^{-1}$ and 128 with $\epsilon = 1.5\text{km}^{-1}$. We can find that with a higher ϵ value, the demand expectation is much closer to the real one. In Figure 25, the relationship between ϵ value and the demand distribution is not obvious in Xi'an. The expectations of the two regions in Xi'an are approximately 69 without geo-indistinguishability, 63 with $\epsilon = 1\text{km}^{-1}$ and 64 with $\epsilon = 1.5\text{km}^{-1}$. We can observe that the expectation value with higher ϵ value is slightly closer to the real value. The differences between the real demand expectations and the expectations with geo-indistinguishability also have influences on performances of revenue. In Figure 26 and 27, we show the performance comparisons with the privacy budget ϵ set to 1 and 1.5km^{-1} under three different metrics. Each data point is the average total revenue and the vertical error bar of each data point is calculated by the standard deviation. We can observe that the average total revenue without preserving privacy is closer to the optimal value than the revenue with differentially private noise. Moreover, with a higher ϵ value, the average revenue is higher and closer to the average revenue without adding noise. Furthermore, the value of error bar (i.e. standard deviation) reduces with more historical data. In the random scheduling method, the standard deviation is higher and the mean total revenue is smaller when we have enough historical data. We can also find that although with a smaller size of historical data, the objective values calculated under FM metric and K metric are still very close to the optimal value. As described in Subsection 7.4.2, the distribution distance measures of FM metric and K metric are tighter than the distance measure of U metric. Furthermore, in Figure 26, the revenue with $\epsilon = 1.5\text{km}^{-1}$ is almost equal to the revenue without privacy protection under three metrics. However, in Figure 27, there is distance between the revenue with $\epsilon = 1.5\text{km}^{-1}$ and the revenue without privacy protection, which is because of the expectations of the real demand distributions and the privacy protected demand distribution.

7.6 Conclusion

In this work, we integrate the geo-indistinguishability scheme based on differential privacy technology to obscure TNC users' location in TNC services and formulate a revenue maximization problem. Because of the uncertainty of the demand of hailing a vehicle in a particular location range, data-driven approach employed to optimize the problem based on the obfuscated location data. In addition, we develop an algorithm to feasibly solve the proposed problem. We conduct simulations on the public real-world dataset from Didi to show the effectiveness of the proposed scheme and illustrate the trade-off between privacy and utility.

8 Future Work

In my future research, I will continue my investigation on cybersecurity, data analysis and machine learning with emphasis on real world applications. I expect to transfer part of my research outcomes into practical applications by building proof-of-concept software and hardware testbeds. With my extensive experience in cybersecurity and machine learning, I plan to conduct future research in the following directions:

- **Investigation on Cyber Threats and Defenses towards Machine Learning Algorithms.** The rising popularity of machine learning-based real world applications is putting the learning pipelines at risk of cyber threats more than ever before. The high capacity of deep neural networks is the main reason behind privacy loss. Sensitive information in the training data can be unintentionally memorized by a deep network. For instance, the medical records contain personal private information like drug usage patterns of the individual patient. Adversarial parties can extract that information given the ability to access or query the network. However, the common strategy of data anonymization is not safe enough because adversarial parties can re-identify individuals in anonymized datasets by combining the data with background information. Under this observation, I plan to explore the potential attacks and the corresponding defense strategies, by investigating these challenges from both theoretical and experimental aspects.

- **Reliable, Secure and Efficient Data Aggregation in Cyber-Physical Systems.** Security is a common key challenge in CPS, since such large-scale systems consist of a large volume of users' daily and sensitive data. Moreover, more and more mobile CPS applications involve users' participation, and users become increasingly concerned about the leakage of their personal information. In fact, there are trade-offs between the effectiveness of privacy protection and the convenience of data aggregation, communications, and energy consumption, which need proper considerations in system designs. Hence, secure data collection and efficient data utilization is the problem I will tackle. For the intermediate entity, I mainly consider how to design the privacy preservation scheme and exploit the knowledge of the private-preserving data in order to achieve reliable accuracy for a wide range of tasks.

- **Location Privacy Preserving Participant Recruitment in Mobile Crowdsensing.** Mobile crowdsensing has emerged as a promising paradigm where location-based sensing tasks are outsourced to mobile participants carrying sensor-equipped devices. A critical issue of crowdsensing is to guarantee the sensing coverage by appropriately recruiting participants, which requires participants' precise locations and thus raises privacy concerns. Based on this motivation, I will study the participant recruitment optimization problem with the consideration of participants' location privacy. Since the privacy preservation scheme may bring uncertainty to the location data, I will develop an uncertainty-aware participant recruitment framework integrating a randomized algorithm with constant-factor approximation guarantee, which can be tolerant of the existence and deletion of biased sensing data incurred by location obfuscation.

Bibliography

[1] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, vol. 19, pp. 1236–1246, November 2018.

[2] A. Rajkomar, E. Oren, K. Chen, A. M. Dai, N. Hajaj, M. Hardt, P. J. Liu, X. Liu, J. Marcus, M. Sun, P. Sundberg, H. Yee, K. Zhang, Y. Zhang, G. Flores, G. E. Duggan, J. Irvine, Q. Le, K. Litsch, A. Mossin, J. Tansuwan, D. Wang, J. Wexler, J. Wilson, D. Ludwig, S. L. Volchenboum, K. Chou, M. Pearson, S. Madabushi, N. H. Shah, A. J. Butte, M. D. Howell, C. Cui, G. S. Corrado, and J. Dean, "Scalable and accurate deep learning with electronic health records," *NPJ Digital Medicine*, vol. 1, p. 18, May 2018.

[3] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. C. Nelson, J. L. Mega, and D. R. Webster, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *Jama*, vol. 316, pp. 2402–2410, December 2016.

[4] A. Yala, C. Lehman, T. Schuster, T. Portnoi, and R. Barzilay, "A deep learning mammography-based model for improved breast cancer risk prediction," *Radiology*, vol. 292, pp. 60–66, May 2019.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, (Montreal, Canada), December 2014.

[6] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015.

[7] M. Fredrikson, S. Jha, and T. Ristenpart, “Model inversion attacks that exploit confidence information and basic countermeasures,” in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, (New York, NY), October 2015.

[8] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *2017 IEEE Symposium on Security and Privacy (SP)*, (San Jose, CA), June 2017.

[9] Z. He, T. Zhang, and R. B. Lee, “Model inversion attacks against collaborative inference,” in *Proceedings of the 35th Annual Computer Security Applications Conference*, (New York, NY), pp. 148–162, December 2019.

[10] J. Liu, H. Shen, H. S. Narman, W. Chung, and Z. Lin, “A survey of mobile crowdsensing techniques: A critical component for the internet of things,” *ACM Transactions on Cyber-Physical Systems*, vol. 2, p. 18, July 2018.

[11] L. Wang, D. Yang, X. Han, T. Wang, D. Zhang, and X. Ma, “Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation,” in *Proceedings of the 26th International Conference on World Wide Web*, (Perth, Australia), April 2017.

[12] M. Qu, H. Zhu, J. Liu, G. Liu, and H. Xiong, “A cost-effective recommender system for taxi drivers,” in *Proceedings of the 20th ACM SIGKDD International Conference*

on Knowledge Discovery and Data Mining, (New York, NY), August 2014.

[13] J. Cramer and A. B. Krueger, "Disruptive change in the taxi business: The case of uber," *American Economic Review*, vol. 106, pp. 177–82, May 2016.

[14] C. Dwork, "Differential privacy: A survey of results," in *International Conference on Theory and Applications of Models of Computation*, (Xi'an, China), April 2008.

[15] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference*, (New York, NY), March 2006.

[16] M. Bun, C. Dwork, G. N. Rothblum, and T. Steinke, "Composable and versatile privacy via truncated cdp," in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, (Los Angeles, CA), June 2018.

[17] A. Rényi, "On measures of entropy and information," in *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, The Regents of the University of California, 1961.

[18] M. Nasr, R. Shokri, and A. Houmansadr, "Machine learning with membership privacy using adversarial regularization," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, (New York, NY), October 2018.

[19] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, (New York, NY), October 2017.

[20] N. Z. Gong and B. Liu, “You are who you know and how you behave: Attribute inference attacks via users’ social friends and behaviors,” in *25th USENIX Security Symposium (USENIX Security 16)*, (Austin, TX), pp. 979–995, August 2016.

[21] N. Z. Gong and B. Liu, “Attribute inference attacks in online social networks,” *ACM Transactions on Privacy and Security (TOPS)*, vol. 21, pp. 1–30, January 2018.

[22] B. Mei, Y. Xiao, R. Li, H. Li, X. Cheng, and Y. Sun, “Image and attribute based convolutional neural network inference attacks in social networks,” *IEEE Transactions on Network Science and Engineering*, vol. 7, pp. 869–879, January 2018.

[23] J. Qian, X.-Y. Li, C. Zhang, and L. Chen, “De-anonymizing social networks and inferring private attributes using knowledge graphs,” in *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, (San Francisco, CA), IEEE, April 2016.

[24] B. Hitaj, G. Ateniese, and F. Pérez-Cruz, “Deep models under the gan: information leakage from collaborative deep learning,” in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, (Dallas, Texas), November 2017.

[25] Y. Chen, S. Wang, D. She, and S. Jana, “On training robust PDF malware classifiers,” in *29th USENIX Security Symposium (USENIX Security 20)*, (Boston, MA), August 2020.

[26] K. Ganju, Q. Wang, W. Yang, C. A. Gunter, and N. Borisov, “Property inference

attacks on fully connected neural networks using permutation invariant representations," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, (Toronto, Canada), October 2018.

[27] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," *arXiv preprint arXiv:1708.06733*, 2017.

[28] M. Jagielski, N. Carlini, D. Berthelot, A. Kurakin, and N. Papernot, "High accuracy and high fidelity extraction of neural networks," in *29th USENIX Security Symposium (USENIX Security 20)*, August 2020.

[29] J. Li, N. Li, and B. Ribeiro, "Membership inference attacks and defenses in supervised learning via generalization gap," *arXiv preprint arXiv:2002.12062*, 2020.

[30] Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, and X. Zhang, "Trojaning attack on neural networks," in *25nd Annual Network and Distributed System Security Symposium, NDSS*, (San Diego, CA), February 2018.

[31] S. J. Oh, B. Schiele, and M. Fritz, "Towards reverse-engineering black-box neural networks," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pp. 121–144, Springer, 2019.

[32] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, (Abu Dhabi, United Arab Emirates), April 2017.

[33] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, (Saarbruecken, Germany), pp. 372–387, IEEE, May 2016.

[34] N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman, "Sok: Security and privacy in machine learning," in *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, (London, UK), April 2018.

[35] A. Salem, R. Wen, M. Backes, S. Ma, and Y. Zhang, "Dynamic backdoor attacks against machine learning models," *arXiv preprint arXiv:2003.03675*, 2020.

[36] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," in *International Conference on Learning Representations*, (Vancouver, Canada), May 2018.

[37] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction apis," in *25th USENIX Security Symposium (USENIX Security 16)*, (Austin, TX), August 2016.

[38] B. Wang and N. Z. Gong, "Stealing hyperparameters in machine learning," in *2018 IEEE Symposium on Security and Privacy (SP)*, (San Francisco, CA), May 2018.

[39] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *2019 IEEE Symposium on Security and Privacy (SP)*, (San Francisco, CA), September 2019.

[40] M. Sabt, M. Achemlal, and A. Bouabdallah, "Trusted execution environment: what it is, and what it is not," in *2015 IEEE Trustcom/BigDataSE/ISPA*, (Helsinki, Finland), December 2015.

[41] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Computing Surveys (CSUR)*, vol. 51, July 2018.

[42] J. Ding, S. M. Errapotu, H. Zhang, Y. Gong, M. Pan, and Z. Han, "Stochastic admm based distributed machine learning with differential privacy," in *International Conference on Security and Privacy in Communication Systems*, (Orlando, FL), December 2019.

[43] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," *arXiv preprint arXiv:1812.00564*, 2018.

[44] E. Li, L. Zeng, Z. Zhou, and X. Chen, "Edge ai: On-demand accelerating deep neural network inference via edge computing," *IEEE Transactions on Wireless Communications*, October 2019.

[45] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[46] R. Detrano, A. Janosi, W. Steinbrunn, M. Pfisterer, J.-J. Schmid, S. Sandhu, K. H. Guppy, S. Lee, and V. Froelicher, "International application of a new probability

algorithm for the diagnosis of coronary artery disease," *The American Journal of Cardiology*, vol. 64, pp. 304–310, August 1989.

[47] P. Suckling J, "The mammographic image analysis society digital mammogram database," *Digital Mammo*, pp. 375–386, 1994.

[48] R. S. Lee, F. Gimenez, A. Hoogi, K. K. Miyake, M. Gorovoy, and D. L. Rubin, "A curated mammography data set for use in computer-aided detection and diagnosis research," *Scientific Data*, vol. 4, p. 170177, December 2017.

[49] A. Hore and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *2010 20th International Conference on Pattern Recognition*, (Istanbul, Turkey), pp. 2366–2369, August 2010.

[50] J. De Fauw, J. R. Ledsam, B. Romera-Paredes, S. Nikolov, N. Tomasev, S. Blackwell, H. Askham, X. Glorot, B. O'Donoghue, and D. Visentin, "Clinically applicable deep learning for diagnosis and referral in retinal disease," *Nature Medicine*, vol. 24, no. 9, p. 1342, 2018.

[51] P. Rajpurkar, J. Irvin, R. L. Ball, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, and C. P. Langlotz, "Deep learning for chest radiograph diagnosis: A retrospective comparison of the chexnext algorithm to practicing radiologists," *PLoS Medicine*, vol. 15, p. e1002686, November 2018.

[52] L. Sweeney, "Only you, your doctor, and many others may know." `https://techscience.org/a/2015092903/`. Accessed: 2019-12-04.

[53] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, (Oxford, UK), July 2018.

[54] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially private empirical risk minimization," *Journal of Machine Learning Research*, vol. 12, pp. 1069–1109, Mar 2011.

[55] D. Wang, C. Chen, and J. Xu, "Differentially private empirical risk minimization with non-convex loss functions," in *International Conference on Machine Learning*, (Long Beach, CA), June 2019.

[56] D. Wang and J. Xu, "Differentially private empirical risk minimization with smooth non-convex loss functions: A non-stationary view," in *Proceedings of the AAAI Conference on Artificial Intelligence*, (Honolulu, HI), January 2019.

[57] J. Ding, X. Zhang, M. Chen, K. Xue, C. Zhang, and M. Pan, "Differentially private robust admm for distributed machine learning," in *2019 IEEE International Conference on Big Data (Big Data)*, (Los Angeles, CA), December 2019.

[58] P. Jain and A. Thakurta, "Differentially private learning with kernels," in *International Conference on Machine Learning*, pp. 118–126, 2013.

[59] J. Ding, X. Zhang, X. Li, J. Wang, R. Yu, and M. Pan, "Differentially private and fair classification via calibrated functional mechanism," in *Proceedings of the AAAI Conference on Artificial Intelligence*, (New York, NY), February 2020.

[60] R. Bassily, A. Smith, and A. Thakurta, "Private empirical risk minimization: Efficient algorithms and tight error bounds," in *Proceedings of the 2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, (Philadelphia, PA), October 2014.

[61] D. Wang, M. Ye, and J. Xu, "Differentially private empirical risk minimization revisited: Faster and more general," in *Advances in Neural Information Processing Systems*, (Long Beach, CA), December 2017.

[62] J. Ding, S. M. Errapotu, H. Zhang, M. Pan, and Z. Han, "Stochastic admm based distributed machine learning with differential privacy," in *International Conference on Security and Privacy in Communication Systems*, (Orlando, FL), October 2019.

[63] B. Wu, S. Zhao, G. Sun, X. Zhang, Z. Su, C. Zeng, and Z. Liu, "P3sgd: Patient privacy preserving sgd for regularizing deep cnns in pathological image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (Long Beach, CA), June 2019.

[64] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, (Vienna, Austria), October 2016.

[65] L. Yu, L. Liu, C. Pu, M. E. Gursoy, and S. Truex, "Differentially private model publishing for deep learning," in *2019 IEEE Symposium on Security and Privacy (SP)*, (San Francisco, CA), May 2019.

[66] L. Wei, B. Luo, Y. Li, Y. Liu, and Q. Xu, "I know what you see: Power side-channel attack on convolutional neural network accelerators," in *Proceedings of the*

34th Annual Computer Security Applications Conference, (San Juan, Puerto Rico), December 2018.

[67] L. Xiang, J. Yang, and B. Li, "Differentially-private deep learning from an optimization perspective," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, (Paris, France), April 2019.

[68] L. Wang, B. Jayaraman, D. Evans, and Q. Gu, "Efficient privacy-preserving nonconvex optimization," *arXiv preprint arXiv:1910.13659*, 2019.

[69] N. Agarwal and K. Singh, "The price of differential privacy for online learning," in *Proceedings of the 34th International Conference on Machine Learning*, (Sydney, Australia), August 2017.

[70] Y. Wang, J. Sharpnack, A. J. Smola, and R. J. Tibshirani, "Learning with differential privacy: Stability, learnability and the sufficiency and necessity of erm principle," *Journal of Machine Learning Research*, vol. 17, no. 183, pp. 1–40, 2016.

[71] S. P. Kasiviswanathan, K. Nissim, and H. Jin, "Private incremental regression," in *Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, (Chicago, IL), May 2017.

[72] P. Jain and A. G. Thakurta, "(Near) dimension independent risk bounds for differentially private learning," in *Proceedings of the 31st International Conference on Machine Learning*, (Beijing, China), June 2014.

[73] D. Kifer, A. Smith, and A. Thakurta, "Private convex empirical risk minimization and high-dimensional regression," in *Proceedings of the 25th Annual Conference on Learning Theory*, (Edinburgh, Scotland), June 2012.

[74] A. Smith and A. G. Thakurta, “Differentially private feature selection via stability arguments, and the robustness of the lasso,” in *Proceedings of the 26th Annual Conference on Learning Theory*, (Princeton, NJ), June 2013.

[75] S. P. Kasiviswanathan and H. Jin, “Efficient private empirical risk minimization for high-dimensional learning,” in *Proceedings of The 33rd International Conference on Machine Learning*, (New York, NY), June 2016.

[76] J. Ding, Y. Gong, C. Zhang, M. Pan, and Z. Han, “Optimal differentially private ADMM for distributed machine learning,” *CoRR*, vol. abs/1901.02094, February 2019.

[77] J. Ding, J. Wang, G. Liang, J. Bi, and M. Pan, “Towards plausible differentially private admm based distributed machine learning,” in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, (New York, NY), October 2020.

[78] J. Lee and D. Kifer, “Concentrated differentially private gradient descent with adaptive per-iteration privacy budget,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (London, United Kingdom), July 2018.

[79] N. Papernot, M. Abadi, Ú. Erlingsson, I. Goodfellow, and K. Talwar, “Semi-supervised knowledge transfer for deep learning from private training data,” in *The 5th International Conference on Learning Representations*, (Toulon, France), April 2017.

[80] L. Xie, K. Lin, S. Wang, F. Wang, and J. Zhou, “Differentially private generative adversarial network,” *arXiv preprint arXiv:1802.06739*, 2018.

[81] X. Zhang, J. Ding, S. M. Errapotu, X. Huang, P. Li, and M. Pan, “Differentially private functional mechanism for generative adversarial networks,” in *2019 IEEE Global Communications Conference (GLOBECOM)*, (Waikoloa, HI), December 2019.

[82] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren, and A. Y. Ng, “Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, (Honolulu, HI), January 2019.

[83] Y. LeCun, C. Cortes, and C. Burges, “MNIST handwritten digit database,” *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.

[84] N. Papernot, A. Thakurta, S. Song, S. Chien, and Ú. Erlingsson, “Tempered sigmoid activations for deep learning with differential privacy,” *arXiv preprint arXiv:2007.14191*, 2020.

[85] A. Krizhevsky and G. Hinton, “Learning multiple layers of features from tiny images (technical report),” *Tech Report*, 2009.

[86] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in *International Conference on Machine Learning*, (Australia), August 2017.

[87] N. Papernot, M. Abadi, U. Erlingsson, I. Goodfellow, and K. Talwar, “Semi-supervised knowledge transfer for deep learning from private training data,” in *Proceddings of the 5th International Conference on Learning Representations (ICLR 2017)*, (Toulon, France), April 2017.

[88] N. Papernot, P. McDaniel, A. Sinha, and M. Wellman, "Towards the science of security and privacy in machine learning," *arXiv preprint arXiv:1611.03814*, 2016.

[89] J. Zhang, Z. Zhang, X. Xiao, Y. Yang, and M. Winslett, "Functional mechanism: regression analysis under differential privacy," *Proceedings of the VLDB Endowment*, vol. 5, pp. 1364–1375, July 2012.

[90] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends in Theoretical Computer Science*, vol. 9, pp. 211–407, August 2014.

[91] C. Dwork, G. N. Rothblum, and S. Vadhan, "Boosting and differential privacy," in *IEEE 51st Annual Symposium on Foundations of Computer Science*, (Las Vegas, NV), October 2010.

[92] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, (Barcelona, Spain), December 2016.

[93] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE Communications Magazine*, vol. 49, pp. 32–39, November 2011.

[94] Z. Wang, J. Hu, R. Lv, J. Wei, Q. Wang, D. Yang, and H. Qi, "Personalized privacy-preserving task allocation for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 18, pp. 1330–1341, June 2019.

[95] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of Risk*, vol. 2, pp. 21–42, April 2000.

[96] D. Yang, G. Xue, X. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM Transactions on Networking (TON)*, vol. 24, pp. 1732–1744, June 2016.

[97] M. Karaliopoulos, O. Telelis, and I. Koutsopoulos, "User recruitment for mobile crowdsensing over opportunistic networks," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, (Kowloon, Hong Kong), April 2015.

[98] W. Tong, J. Hua, and S. Zhong, "A jointly differentially private scheduling protocol for ridesharing services," *IEEE Transactions on Information Forensics and Security*, vol. 12, pp. 2444–2456, May 2017.

[99] X. Zhang, J. Wang, Y. Li, R. Jäntti, M. Pan, and Z. Han, "Catching all pokémon: Virtual reward optimization with tensor voting based trajectory privacy," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 883–892, January 2019.

[100] H. Liu, X. Li, H. Li, J. Ma, and X. Ma, "Spatiotemporal correlation-aware dummy-based privacy protection scheme for location-based services," in *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*, (Atlanta, GA), May 2017.

[101] S. Hayashida, D. Amagata, T. Hara, and X. Xie, "Dummy generation based on user-movement estimation for location privacy protection," *IEEE Access*, vol. 6, pp. 22958–22969, April 2018.

[102] R. Shokri, G. Theodorakopoulos, P. Papadimitratos, E. Kazemi, and J.-P. Hubaux, "Hiding in the mobile crowd: Location privacy through collaboration," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, pp. 266–279, May 2014.

[103] T. Peng, Q. Liu, D. Meng, and G. Wang, "Collaborative trajectory privacy preserving scheme in location-based services," *Information Sciences*, vol. 387, pp. 165–179, 2017.

[104] N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Optimal geo-indistinguishable mechanisms for location privacy," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS'14)*, (Scottsdale, AZ), November 2014.

[105] X. Liu, H. Zhao, M. Pan, H. Yue, X. Li, and Y. Fang, "Traffic-aware multiple mix zone placement for protecting location privacy," in *Proceeding of IEEE International Conference on Computer Communications (INFOCOM)*, (Orlando, FL), March 2012.

[106] I. Memon, Q. A. Arain, M. H. Memon, F. A. Mangi, and R. Akhtar, "Search me if you can: Multiple mix zones with location privacy protection for mapping services," *International Journal of Communication Systems*, vol. 30, p. e3312, April 2017.

[107] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Computing*, vol. 2, pp. 46–55, January-March 2003.

[108] K. Vu, R. Zheng, and J. Gao, "Efficient algorithms for k-anonymous location privacy in participatory sensing," in *Proceeding of IEEE International Conference on Computer Communications (INFOCOM)*, (Orlando, FL), March 2012.

[109] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, (Berkeley, CA), October 2013.

[110] K. Chatzikokolakis, M. E. Andrés, N. E. Bordenabe, and C. Palamidessi, "Broadening the scope of differential privacy using metrics," in *International Symposium on Privacy Enhancing Technologies Symposium*, (Bloomington, IN), July 2013.

[111] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security (CCS'13)*, (New York, NY), November 2013.

[112] K. Chatzikokolakis, C. Palamidessi, and M. Stronati, "A predictive differentially-private mechanism for mobility traces," in *International Symposium on Privacy Enhancing Technologies Symposium*, (Amsterdam, The Netherlands), July 2014.

[113] M. R. Gary and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness.* WH Freeman and Company, New York, 1979.

[114] M. Pan, P. Li, Y. Song, Y. Fang, and P. Lin, "Spectrum clouds: A session based spectrum trading system for multi-hop cognitive radio networks," in *Proceeding of IEEE International Conference on Computer Communications (INFOCOM)*, (Orlando, FL), March 2012.

[115] X. Li, Y. Sun, Y. Guo, X. Fu, and M. Pan, "Dolphins first: Dolphin-aware communications in multi-hop underwater cognitive acoustic networks," *IEEE Transactions on Wireless Communications*, vol. 16, pp. 2043–2056, April 2017.

[116] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear Programming and Network Flows.* John Wiley & Sons, 2011.

[117] L. Bracciale, M. Bonola, P. Loreti, G. Bianchi, R. Amici, and A. Rabuffi, "CRAWDAD dataset roma/taxi (v. 2014-07-17)." Downloaded from `https://crawdad.org/roma/taxi/20140717`, July 2014.

[118] X. Gu, M. Li, Y. Cao, and L. Xiong, "Supporting both range queries and frequency estimation with local differential privacy," in *2019 IEEE Conference on Communications and Network Security (CNS)*, (Washington DC, DC), August 2019.

[119] N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, and G. Yu, "Collecting and analyzing multidimensional data with local differential privacy," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, (Macau, China), April 2019.

[120] Y. Xiao and L. Xiong, "Protecting locations with differential privacy under temporal correlations," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS'15)*, (Denver, CO), October 2015.

[121] C. Zhao and Y. Guan, "Data-driven risk-averse two-stage stochastic program with ζ-structure probability metrics," *Available on Optimization Online*, 2015.

[122] Y. Huang, F. Bastani, R. Jin, and X. S. Wang, "Large scale real-time ridesharing with service guarantee on road networks," *Proceedings of the VLDB Endowment*, vol. 7, pp. 2017–2028, October 2014.

[123] S. Ma, Y. Zheng, and O. Wolfson, "T-share: A large-scale dynamic taxi ridesharing service," in *2013 IEEE 29th International Conference on Data Engineering (ICDE)*, (Brisbane, Australia), April 2013.

[124] A. Y. Lam, Y.-W. Leung, and X. Chu, "Autonomous-vehicle public transportation system: scheduling and admission control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, pp. 1210–1226, January 2016.

[125] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Enhancing privacy through caching in location-based services," in *Proceeding of IEEE International Conference on Computer Communications (INFOCOM)*, (Hong Kong, China), April 2015.

[126] S. Mascetti, L. Bertolaja, and C. Bettini, "A practical location privacy attack in proximity services," in *2013 IEEE 14th International Conference on Mobile Data Management*, (Milan, Italy), June 2013.

[127] S. Mascetti, L. Bertolaja, and C. Bettini, "Safebox: Adaptable spatio-temporal generalization for location privacy protection," *Transactions on Data Privacy*, vol. 7, pp. 131–163, August 2014.

[128] A. Monreale, W. H. Wang, F. Pratesi, S. Rinzivillo, D. Pedreschi, G. Andrienko, and N. Andrienko, "Privacy-preserving distributed movement data aggregation," in *Geographic Information Science at the Heart of Europe*, pp. 225–245, Springer, 2013.

[129] G. C. Calafiore, "Ambiguous risk measures and optimal robust portfolios," *SIAM Journal on Optimization*, vol. 18, pp. 853–877, October 2007.

[130] A. M. Vershik, "Kantorovich metric: Initial history and little-known applications," *Journal of Mathematical Sciences*, vol. 133, pp. 1410–1417, March 2006.

[131] H. Heitsch and W. Römisch, "A note on scenario reduction for two-stage stochastic programs," *Operations Research Letters*, vol. 35, pp. 731–738, January 2007.

[132] A. Dvoretzky, J. Kiefer, and J. Wolfowitz, “Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator,” *The Annals of Mathematical Statistics*, pp. 642–669, 1956.

[133] “Didi chuxing gaia open dataset initiative.” `https://gaia.didichuxing.com`.