Automatic Tuning of Privacy Budgets in Input-Discriminative Local Differential Privacy

Takao Murakami, *Member, IEEE*, and Yuichi Sei, *Member, IEEE*

***Abstract*—Local differential privacy (LDP) and its variants have been recently studied to analyze personal data collected from Internet of Things (IoT) devices while strongly protecting user privacy. In particular, a recent study proposes a general privacy notion called input-discriminative LDP (ID-LDP), which introduces a privacy budget for each input value to deal with different levels of sensitivity. However, it is unclear how to set an appropriate privacy budget for each input value, especially, in current situations where reidentification is considered a major risk, e.g., in GDPR. Moreover, the possible number of input values can be very large in IoT. Consequently, it is also extremely difficult to manually check whether a privacy budget for each input value is appropriate. In this article, we propose algorithms to automatically tune privacy budgets in ID-LDP so that obfuscated data strongly prevent reidentification. We also propose a new instance of ID-LDP called one-budget ID-LDP (OneID-LDP) to prevent reidentification with high utility. Through comprehensive experiments using four real data sets, we show that existing instances of ID-LDP lack either utility or privacy—they overprotect personal data or are vulnerable to reidentification attacks. Then, we show that our OneID-LDP mechanisms with our privacy budget tuning algorithm provide much higher utility than LDP mechanisms while strongly preventing reidentification.**

***Index Terms*—Frequency estimation, Internet of Things (IoT), local differential privacy (LDP), privacy budget, reidentification.**

I. Introduction

With the advancement of Internet of Things (IoT) devices, such as wearable devices, connected cars, smart homes, and activity monitoring systems, personal data are increasingly collected for various types of data analysis. For example, a large amount of location data collected from wearable devices or connected cars are analyzed to calculate a frequency distribution (geographic population distribution). The frequency distribution is useful for providing traffic information to users [1] or finding popular point of interests (POIs), such as restaurants and cultural landmarks [2]. For another example, person activity data from monitoring systems are analyzed to extract typical activity patterns of elderly people [3]. Power-consumption data from smart meters are analyzed to find typical daily consumption patterns [4] or the right customers to target for demand response programs [5]. Although these data are useful for industry and society, the disclosure of personal data can lead to serious privacy issues. Therefore, there is a need to develop algorithms to perform data analysis while strongly protecting user privacy.

Differential privacy (DP) [6], [7] is known as a gold standard for a private analysis. It strongly protects user privacy against adversaries with any background knowledge. There are roughly two types of DP: 1) centralized DP and 2) local DP (LDP). Centralized DP assumes a centralized model where a central server has personal data of all users and obfuscates analysis results, e.g., frequency distribution. In this model, there is a risk that the personal data of all users are leaked from the server by illegal access [8]. In contrast, LDP assumes a local model where each user obfuscates her personal data and sends the obfuscated data to a data collector; i.e., it does not assume a trusted party. Thus, LDP does not suffer from the data breach issue and has been adopted in companies, such as Google [9], Apple [10], and Microsoft [11].

LDP prevents an adversary from distinguishing any pair of input values and controls the indistinguishability by a parameter called a *privacy budget* ε. LDP regards all input values as equally sensitive and uses the same privacy budget for all pairs of input values. However, different input values have different levels of sensitivity in practice. For example, homes and hospitals are highly sensitive locations, whereas parks, restaurants, and sightseeing places would be less sensitive for most users. Cancers and HIV are highly sensitive diseases, whereas headache, sore throat, and stomachache would be less sensitive. In these scenarios, LDP mechanisms excessively obfuscate personal data and cause the loss of data utility.

To address this issue, a recent study proposed a general privacy notion called input-discriminative LDP (ID-LDP) [12]. ID-LDP deals with different levels of sensitivity in input values by introducing a privacy budget ε_x for each input value x. ID-LDP controls the indistinguishability of a pair of two input values x and x' as a function of the corresponding budgets ε_x and $\varepsilon_{x'}$. ID-LDP is general in that we can use any function for the pair. It includes MinID-LDP [12] and high-low LDP (HLLDP) [13], [14] as instances, both of which provide higher utility than LDP.

However, it is difficult to manually determine an appropriate privacy budget for each input value in practice. For example, it is well known that DP strongly protects user privacy when the privacy budget is small; e.g., $\varepsilon \leq 1$ [15]. Thus, it is natural

Manuscript received 11 January 2022; revised 26 September 2022; accepted 10 April 2023. Date of publication 14 April 2023; date of current version 7 September 2023. This work was supported in part by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant 19H04113 and Grant 21H03496. *(Takao Murakami and Yuichi Sei contributed equally to this work.) (Corresponding authors: Takao Murakami; Yuichi Sei.)*

Takao Murakami is with the Cyber Physical Security Research Center, National Institute of Advanced Industrial Science and Technology, Tokyo 135-0064, Japan (e-mail: takao-murakami@aist.go.jp).

Yuichi Sei is with the Graduate School of Informatics and Engineering, University of Electro-Communications, Tokyo 1828585, Japan (e-mail: seiuny@uec.ac.jp).

Digital Object Identifier 10.1109/JIOT.2023.3267082

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

to allocate such small privacy budgets to sensitive locations, such as homes and hospitals. However, it is unclear how much privacy budgets should be allocated to less sensitive locations, such as parks and restaurants.

In particular, even if some input values are nonsensitive for users, the disclosure of them may lead to the *reidentification* of records [16]. For example, assume that Alice disclosed the fact that she went to a coffee shop, which was nonsensitive for her. An adversary who obtains this information may use it for reidentifying another sensitive record (e.g., hospital she regularly visits near the coffee shop) in a different database. Consequently, various kinds of personal data from different databases may be linked to make a *user profile* [17], and it might be sold on the dark Web [18]. Since reidentification is considered a major risk in general data protection regulation (GDPR) [19], [20], we need to strongly protect all personal data, including nonsensitive data, from reidentification attacks.

Moreover, IoT devices can collect various data, and the possible number of input values can also be very large, e.g., larger than 10 000 in our experiments. Thus, it is extremely difficult to manually set an appropriate privacy budget for each input value and to manually check whether each value is appropriate. Setting an appropriate privacy budget is also recognized as an important challenge in Internet of Vehicles (IoV) [21].

In this article, we propose algorithms to *automatically* determine privacy budgets in ID-LDP so that obfuscated data strongly prevent reidentification. Note that in many practical scenarios of the local model, an adversary needs to perform reidentification attacks to link the obfuscated data to users [22]. For example, some applications (e.g., Foursquare, Google Maps, and YouTube recommendation) can be used without a login, i.e., without sending a user ID. For another example, a data collector pseudonymizes obfuscated data to reduce the risks to the users, as described in GDPR [19]. In both cases, an outsider adversary who obtains the obfuscated data needs to reidentify the data. Our algorithms automatically determine privacy budgets to strongly prevent this attack.

As a task for the data collector, we consider *frequency estimation* [9], [12], [23], which is a fundamental task in the local model. We show that our proposed algorithms strongly prevent the reidentification attack while providing much higher utility than LDP.

Our Contributions: Our contributions are as follows.

1) We propose *privacy budget tuning algorithms* for ID-LDP, which automatically determine privacy budgets so that obfuscated data prevent reidentification. To our knowledge, this work is the first to automatically determine a privacy budget for each input value to prevent reidentification (see Section II for details).
2) We also propose a new instance of ID-LDP called one-budget ID-LDP (OneID-LDP) to bound the reidentification risk with high utility. We prove that OneID-LDP upper bounds the reidentification accuracy for every obfuscated data, hence, every user.
3) Through comprehensive experiments using four real data sets (one location data set with six cities and three person activity data sets), we show that two existing instances of ID-LDP lack either utility or privacy—MinID-LDP [12] still overprotects personal data and lacks utility, and HLLDP [13], [14] is vulnerable to reidentification.
4) Finally, we show the effectiveness of our algorithms using the four data sets. Specifically, we show that our OneID-LDP mechanisms with our privacy budget tuning algorithms provide much higher utility than MinID-LDP and LDP mechanisms while preventing reidentification.

TABLE I
RELATIONSHIP BETWEEN THE EXISTING WORK AND OUR PROPOSAL. OUR PROPOSAL IS HIGHLIGHTED IN BOLD

	Manual setting of privacy budgets	**Automatic tuning of privacy budgets**
HLLDP	[13], [14]	N/A
MinID-LDP	[12]	**Sections IV-E and IV-F**
OneID-LDP	**Section IV-B**	**Sections IV-D and IV-F**

TABLE II
PRIVACY AND UTILITY OF THE THREE PRIVACY NOTIONS. WE ASSUME THAT OUR PRIVACY BUDGET TUNING ALGORITHMS ARE APPLIED TO MINID-LDP AND ONEID-LDP

	Privacy against re-identification	Utility
HLLDP	Low	High
MinID-LDP	High	Low
OneID-LDP	High	High

Novelty: Below, we explain the novelty of our work in more detail. As explained above, our proposal is twofold: 1) *automatic tuning of privacy budgets* and 2) *OneID-LDP*. Table I shows the relationship between the existing work and our proposal.

First and most importantly, the automatic tuning of privacy budgets (i.e., the third column of Table I) is a totally new research direction. All of the existing work on ID-LDP [12], [13], [14] manually set a privacy budget ε_x for each input value x without theoretical justification; e.g., $\varepsilon_x = \ln 6$ [12] or ∞ [13], [14] for nonsensitive data. In contrast, our privacy budget tuning algorithms automatically determine ε_x to provide theoretical guarantees against reidentification attacks.

Second, our OneID-LDP (i.e., the fourth row of Table I) is a new privacy notion. OneID-LDP is designed to prevent reidentification with much higher utility than MinID-LDP. We propose OneID-LDP with a manual setting of ε_x in Section IV-B. Then, we propose privacy budget tuning algorithms for OneID-LDP (resp., MinID-LDP) in Sections IV-D and IV-F (resp., Sections IV-E and IV-F). We show that both OneID-LDP and MinID-LDP prevent reidentification when using our privacy budget tuning algorithms. Then, we show that OneID-LDP provides much higher utility than MinID-LDP.

Note that our privacy budget tuning algorithms cannot be applied to HLLDP ("N/A" in Table I). This is because HLLDP always sets $\varepsilon_x = \infty$ for nonsensitive data. In contrast, our privacy budget tuning algorithms use a finite value of ε_x for some nonsensitive data to prevent reidentification. Thus, they are incompatible with HLLDP.

Table II summarizes the privacy and utility of the three privacy notions. Here, we apply our privacy budget tuning algorithms to MinID-LDP and OneID-LDP. We say an algorithm provides "high privacy against reidentification" when it upper bounds the reidentification accuracy for every user by a desired value. Because HLLDP does not protect nonsensitive data at all (i.e., $\varepsilon_x = \infty$), it is vulnerable to reidentification. MinID-LDP lacks utility. In contrast, our OneID-LDP provides high privacy and utility. See Section V for details.

Remark on Privacy Risks: This article shows that our proposal (OneID-LDP with automatic tuning of privacy budgets) is secure against reidentification. There are other privacy risks in the privacy literature. Specifically, two types of information disclosure are known as privacy risks: 1) *identity disclosure* and 2) *attribute disclosure* [24]. Identity disclosure takes place when the adversary correctly links a user to a record in the database. Attribute disclosure takes place when the adversary correctly obtains some information about an attribute of a user.

Identity disclosure is caused by *reidentification attacks* and *membership inference attacks* [25], [26] as follows. The adversary first performs membership inference attacks, which determine who are members, i.e., users in the database. Then, the adversary performs reidentification attacks, which link one of the (inferred) members to each record in the database. The adversary succeeds in identity disclosure if she accurately performs both membership inference and reidentification. In this article, we assume that the adversary completely knows who are members/nonmembers when she performs reidentification. In other words, we consider a *worst case* scenario where the accuracy of membership inference is 100%. In practice, the accuracy of membership inference would be smaller than 100%. In that case, the accuracy of identity disclosure would be smaller than what is reported in our experiments.

Attribute disclosure is caused by *attribute inference attacks* [27], which infer an attribute of a user. LDP is a privacy notion to strongly prevent the inference of attributes from output data. Similarly, our OneID-LDP strongly prevents the inference of sensitive attributes from output data. One might think that the adversary might infer attributes from a frequency distribution. For example, assume that users in a certain area are likely to visit a hospital and that Alice lives in this area. Then, the adversary who obtains a frequency distribution estimated by the data collector would infer that Alice is likely to visit the hospital. This kind of attack is inevitable in *any* LDP (or ID-LDP) mechanism when the goal is to estimate a frequency distribution. In addition, this kind of inference is *not* considered a privacy violation in [28], because it is statistical inference. Thus, it is outside the scope of this article.

In summary, our proposal strongly prevents both identity and attribute disclosure other than statistical inference.

Paper Organization: The remainder of this article is organized as follows. In Section II, we review the previous work related to ours. In Section III, we explain some preliminaries for our work, such as basic notations, utility/privacy metrics, and randomized mechanisms. In Section IV, we propose our privacy budget tuning algorithms and OneID-LDP. We also prove that both OneID-LDP and MinID-LDP bound the reidentification accuracy when using our privacy budget tuning algorithms. In Section V, we show our experimental results. In Section VI, we conclude this article.

II. Related Work

LDP and Variants: LDP [29], a local model version of DP, has been widely studied in both academia [9], [23], [30], [31], [32], [33], [34], [35], [36], [37] and industry [9], [10], [11]. LDP has also been applied to IoT, such as IoV [21], [37], wearable sensors [38], and blockchain-based IoT [39], [40].

The limitation of LDP is that it requires too much noise; it is proven in [32] that LDP needs an extremely large number of users (e.g., dozen million [9]) to enable accurate data analysis due to the large noise. One reason for the low utility of LDP is that it regards all input data as equally sensitive.

Numerous variants of DP/LDP have been studied to overcome its limitation. A recent Systems of Knowledge (SoK) paper [41] classifies these variants into seven categories, depending on which aspect of the original DP/LDP is modified. Out of the seven categories, we focus on the **V** (**V**ariation of Privacy Loss) category because it attempts to address the utility issue explained above; see [41] for details of the other six categories. The **V** category varies the privacy level of DP/LDP across inputs. The variants in this category include HLLDP [13], [14], MinID-LDP [12], and context-aware LDP [14].

The first variant of LDP in the **V** category was proposed by Murakami and Kawamoto [13]. They introduced the notion of HLLDP,[1] which provides a privacy guarantee equivalent to LDP only for sensitive data. Then, they proposed a subclass of HLLDP called utility-optimized LDP (ULDP), which optimizes the utility within HLLDP. Later, Gu et al. [12] proposed the notion of ID-LDP that includes HLLDP as a special case. They proposed an instance of ID-LDP called MinID-LDP and showed that it provides higher utility than LDP. In this article, we focus on ID-LDP because it is a general notion—it includes both HLLDP and MinID-LDP as instances.

Another interesting variant of LDP in the **V** category is context-aware LDP proposed by Acharya et al. [14]. It allocates a privacy budget $\varepsilon_{x,x'}$ for each pair of two input values x and x'. This is also very general and includes various variants of LDP as special cases, e.g., HLLDP, geo-indistiguishability [43], and d_x-privacy [44]. They also introduced a new instance of context-aware LDP called block-structured LDP [14], which hides input values within the same group. We do not focus on context-aware LDP, because our interest is in handling different levels of sensitivity in input values, as described in Section I. For this purpose, it is sufficient to use ID-LDP that allocates a privacy budget ε_x to each personal data x.

A crucial issue in these variants of LDP is how to set appropriate privacy budgets. As explained in Section I, the disclosure of nonsensitive input values may lead to the reidentification of records in another database. It is extremely difficult to manually set an appropriate privacy budget for each input value, as the possible number of input values can be very large. Unfortunately, all of the above studies [12], [13], [14] do not consider how to automatically set appropriate privacy budgets. Therefore, we propose privacy budget tuning algorithms

[1] Murakami and Kawamoto [13] called this privacy notion one-sided LDP (OSLDP) because it is a local model version of one-sided DP (OSDP) [42].

and a new instance of ID-LDP called OneID-LDP to prevent reidentification while keeping high utility.

DP and Reidentification Risks: The relationship between DP and the reidentification risk was shown in recent studies by Cohen and Nissim [45], [46]. Specifically, they formally defined a concept of predicate singling out, which is weaker than singling out in GDPR. Security against predicate singling out is a necessary (but not sufficient) condition for security against singling out in GDPR. They showed that DP prevents predicate singling out (whereas k-anonymity does not) in an asymptotic setting where the number of users goes to ∞. However, they do not clarify the relationship between the privacy budget in DP and the reidentification risk. They also do not consider different levels of sensitivity in input values.

There are also some variants of DP related to the reidentification risk. For example, Gehrke et al. [47] proposed the notion of crowd-blending privacy, which weakens centralized DP so that each record is indistinguishable from at least $k-1$ other records. Bindshaedler et al. [48] proposed a similar notion called plausible deniability and showed that a plausible deniability mechanism generates differentially private synthetic data under some conditions. Murakami and Takahashi [22] proposed personal information entropy (PIE) privacy as a relaxation of LDP to reduce the reidentification risk.[2] All of these studies [22], [47], [48] do not consider different levels of sensitivity in input values.

Finally, we note that our work is totally different from a recently proposed shuffling technique [49], [50], [51]. Specifically, the shuffling technique reduces the privacy budget in DP by introducing an intermediate server (shuffler) that randomly shuffles obfuscated data. Our work is different from this technique in three ways. First, our algorithms upper bound the reidentification accuracy, whereas the shuffling technique does not consider it. Second, our work deals with different levels of sensitivity in input values, whereas the shuffling technique does not. Third, we show that reidentification is strongly prevented even if the privacy budget is ∞ in some cases (see Section IV-D for details), whereas the shuffling technique cannot reduce the privacy budget in this case.

In summary, our work is the first to automatically determine a privacy budget for each input value to prevent reidentification, to our knowledge.

III. Preliminaries

In this section, we provide some preliminaries for our work. Section III-A introduces basic notations used in this article. Sections III-B–III-D explain utility metrics, privacy metrics, and randomized mechanisms, respectively.

A. Notations

Let $\mathbb{R}$, $\mathbb{N}$, $\mathbb{R}_{\geq 0}$, $\mathbb{Z}_{\geq 0}$ be the sets of real numbers, natural numbers, nonnegative real numbers, and nonnegative integers, respectively. For $a \in \mathbb{N}$, let $[a] = \{1, 2, \ldots, a\}$. All logarithms in this article are base e.

TABLE III
Basic Notations

Symbol	Description
n	Number of users.
$\mathcal{U}$	Finite set of users ($\mathcal{U} = \{u_1, \cdots, u_n\}$).
u_i	i-th user.
$\mathcal{X}$	Finite set of personal data.
$X^{(i)}$	Random variable representing personal data of user u_i.
$\mathbf{X}$	Finite set of all personal data ($\mathbf{X} = \{X^{(1)}, \cdots, X^{(n)}\}$).
$\mathcal{Y}$	Finite set of obfuscated data.
$Y^{(i)}$	Random variable representing obfuscated data of user u_i.
$\mathbf{Y}$	Finite set of all obfuscated data ($\mathbf{Y} = \{Y^{(1)}, \cdots, Y^{(n)}\}$).
$\mathbf{Q}$	Randomized mechanism.
$\mathbf{c}$	Frequency distribution of personal data.
$\hat{\mathbf{c}}$	Estimate of $\mathbf{c}$.

Let $\mathcal{U}$ be a finite set of users who use an application (e.g., wearable device and connected car). Let $n \in \mathbb{N}$ be the number of users, and $u_i \in \mathcal{U}$ be the ith user; i.e., $\mathcal{U} = \{u_1, \ldots, u_n\}$.

Let $\mathcal{X}$ be a finite set of personal data (e.g., locations and physical activities). We assume that continuous data are discretized into some bins; e.g., a location map is divided into smaller regions or POIs. We also assume that each user u_i sends a single datum (we discuss the case where a user sends multiple data in Section IV-B). Let $X^{(i)}$ be a random variable representing personal data of user u_i. Let $\mathbf{X} = \{X^{(1)}, \ldots, X^{(n)}\}$ be a set of all personal data.

Let $\mathcal{Y}$ be a finite set of obfuscated data. Let $Y^{(i)}$ be a random variable representing obfuscated data of user u_i. Let $\mathbf{Y} = \{Y^{(1)}, \ldots, Y^{(n)}\}$ be a set of all personal data. Each user obfuscates her personal data using a randomized mechanism $\mathbf{Q}$, which maps $x \in \mathcal{X}$ to $y \in \mathcal{Y}$ with probability $\mathbf{Q}(y|x)$, and sends the obfuscated data to a data collector.

We consider frequency estimation as a task of the data collector. Let $\mathbf{c}$ be a frequency distribution, whose element $\mathbf{c}(x)$ is the number of users who possess x; i.e.,

$$\mathbf{c}(x) = \sum_{i \in [n]} \mathbf{1}_{X^{(i)}=x}$$

where $\mathbf{1}_{X^{(i)}=x}$ is an indicator function that takes 1 if $X^{(i)} = x$ and 0 otherwise. The data collector estimates a frequency distribution $\mathbf{c}$ from obfuscated data of all users. Let $\hat{\mathbf{c}}$ be an estimate of $\mathbf{c}$.

Table III shows the basic notations in this article.

B. Utility Metrics

In this article, we use the mean absolute error (MAE) and mean-squared error (MSE) as metrics of utility loss. The MAE and MSE are defined using the l_1 loss and l_2 loss, respectively.

Specifically, let l_1 (resp., l_2^2) be the l_1 (resp., l_2) loss function, which maps a frequency distribution $\mathbf{c}$ and its estimate $\hat{\mathbf{c}}$ to the loss; i.e., $l_1(\mathbf{c}, \hat{\mathbf{c}}) = \sum_{x \in \mathcal{X}} |\mathbf{c}(x) - \hat{\mathbf{c}}(x)|$, $l_2^2(\mathbf{c}, \hat{\mathbf{c}}) = \sum_{x \in \mathcal{X}} (\mathbf{c}(x) - \hat{\mathbf{c}}(x))^2$. Then, the MAE is the mean of $l_1(\mathbf{c}, \hat{\mathbf{c}})$ over multiple realizations of obfuscated data $\mathbf{Y}$. The MSE is the mean of $l_2^2(\mathbf{c}, \hat{\mathbf{c}})$ over multiple realizations of $\mathbf{Y}$.

C. Privacy Metrics

LDP: LDP is defined as follows.

[2] We also note that PIE privacy is weak in that some users can be completely reidentified. In contrast, our OneID-LDP guarantees a small reidentification risk for every obfuscated data, hence, every user. In other words, there are no victims in OneID-LDP. See Section IV-D for details.

Definition 1 (ε-LDP [29]): Let $\varepsilon \in \mathbb{R}_{\geq 0}$ be a privacy budget. A randomized mechanism $\mathbf{Q}$ provides *ε-LDP* if and only if for any $x, x' \in \mathcal{X}$ and any $y \in \mathcal{Y}$

$$\mathbf{Q}(y|x) \leq e^{\varepsilon}\mathbf{Q}(y|x').$$

Intuitively, LDP guarantees that an adversary who obtains obfuscated data y cannot determine, for any pair of input values x and x', whether it comes from x or x'. This holds, especially, when the privacy budget ε is close to 0 because all of the input values in $\mathcal{X}$ are almost equally likely; i.e., $\mathbf{Q}(y|x) \approx \mathbf{Q}(y|x')$ for any x and x'. Thus, LDP strongly protects user privacy when ε is small; e.g., $\varepsilon \leq 1$ [15].

ID-LDP: LDP regards all input values in $\mathcal{X}$ as equally sensitive. However, the sensitivity differs according to the input values in practice; e.g., hospitals and homes are highly sensitive locations, whereas other locations, such as parks and restaurants, are not sensitive for most users. Thus, LDP causes excessive obfuscation and a significant loss of utility.

To address this issue, Gu et al. [12] proposed ID-LDP. The feature of ID-LDP is that it introduces a privacy budget ε_x for each input value x in $\mathcal{X}$. Formally, ID-LDP is defined as follows.

Definition 2 ($(\mathcal{E}, r)$-ID-LDP [12]): Let $\varepsilon_x \in \mathbb{R}_{\geq 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$. Let $r : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a function that takes two privacy budgets as input and outputs a nonnegative value. A randomized mechanism $\mathbf{Q}$ provides $(\mathcal{E}, r)$*-ID-LDP* if and only if for any $x, x' \in \mathcal{X}$ and any $y \in \mathcal{Y}$

$$\mathbf{Q}(y|x) \leq e^{r(\varepsilon_x, \varepsilon_{x'})}\mathbf{Q}(y|x'). \tag{1}$$

We refer to $r(\varepsilon_x, \varepsilon_{x'})$ as a *pair budget* for x and x'. ID-LDP is general in that we can use any function as r.

MinID-LDP: As an instance of ID-LDP, Gu et al. [12] proposed MinID-LDP.

Definition 3 ($\mathcal{E}$-MinID-LDP [12]): Let $\varepsilon_x \in \mathbb{R}_{\geq 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$. A randomized mechanism $\mathbf{Q}$ provides *$\mathcal{E}$-MinID-LDP* if and only if it provides $(\mathcal{E}, r)$-ID-LDP, where $r(\varepsilon_x, \varepsilon_{x'}) = \min\{\varepsilon_x, \varepsilon_{x'}\}$.

MinID-LDP controls the adversary's capability of distinguishing x and x' by using the minimum of ε_x and $\varepsilon_{x'}$. For example, assume that the set of personal data is $\mathcal{X} = \{\text{cancer, headache, sore throat}\}$. We set $\varepsilon_{\text{cancer}} = 1$ and $\varepsilon_{\text{headache}} = \varepsilon_{\text{sore throat}} = 2$ because a cancer is the most sensitive disease. Then, MinID-LDP adopts 1 as a pair budget for (cancer, headache) and 2 for (headache, sore throat). For a pair of nonsensitive input values x and x', MinID-LDP can assign a large pair budget. However, it needs to use a small pair budget when either x or x' is sensitive. Consequently, when we consider a reidentification as a risk, MinID-LDP still overprotects personal data—the utility gain of MinID-LDP over LDP is limited, as shown in our experiments.

HLLDP: Murakami and Kawamoto [13] and Acharya et al. [14] introduced HLLDP. HLLDP assigns a privacy budget $\varepsilon_S \in \mathbb{R}_{\geq 0}$ for sensitive data and ∞ for nonsensitive data. Although the relationship between HLLDP and ID-LDP is not clarified in [12], [13], and [14], HLLDP is an instance of ID-LDP. Specifically, we can define HLLDP as follows.

Definition 4 ($(\mathcal{X}_S, \varepsilon_S)$-HLLDP [13], [14]): Let $\varepsilon_x \in \mathbb{R}_{\geq 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$. Let $\mathcal{X}_S \subseteq \mathcal{X}$ be a finite set of sensitive data. Let $\varepsilon_S \in \mathbb{R}_{\geq 0}$ be a privacy budget for sensitive data. A randomized mechanism $\mathbf{Q}$ provides $(\mathcal{X}_S, \varepsilon_S)$-HLLDP if and only if it provides $\mathcal{E}$-ID-LDP, where

$$\varepsilon_x = \begin{cases} \varepsilon_S, & (\text{if } x \in \mathcal{X}_S) \\ \infty, & (\text{otherwise}) \end{cases} \tag{2}$$

and $r(\varepsilon_x, \varepsilon_{x'}) = \varepsilon_x$.

Since HLLDP assigns $\varepsilon_x = \infty$ for nonsensitive data x, it provides much higher utility than LDP [13]. However, this comes at the expense of privacy—HLLDP is vulnerable to reidentification attacks, as shown in our experiments.

Remark on Sensitive Data: Note that the distinction between sensitive and nonsensitive data can be different from user to user; e.g., $x_1 \in \mathcal{X}$ is sensitive for Alice and Bob, whereas $x_2 \in \mathcal{X}$ is sensitive for only Carol. The study in [13] proposes a distribution estimation method under LDP in such a personalized scenario. Specifically, their method first maps sensitive data for each user to a bot symbol "$\perp$" and uses an ID-LDP mechanism with domain $\mathcal{X} \cup \{\perp\}$. After computing the frequency distribution of input data including $\perp$, their method discards the frequency of $\perp$ and normalizes the other frequencies so that the sum is n. It is shown in [13] that a distribution can be accurately estimated by using this method.

In our experiments, we assume that the set of sensitive data is common to all users, e.g., POIs with "home" and "hospital" categories in the location data set. However, our proposed methods are easily extended to the personalized scenario explained above by mapping each user's sensitive data to a bot $\perp$ in the same way as [13].

D. Randomized Mechanisms

UE: As a randomized mechanism $\mathbf{Q}$ providing LDP, we use the unary encoding (UE) mechanism [23]. The set of obfuscated data in UE is $\mathcal{Y} = \{0, 1\}^{|\mathcal{X}|}$.

Specifically, we express the set of personal data as $\mathcal{X} = \{x_1, \dots, x_{|\mathcal{X}|}\}$ without loss of generality. For any $k \in [|\mathcal{X}|]$, the UE mechanism first maps x_k to the kth standard basis vector $\mathbf{e}_k = (0, \dots, 0, 1, 0, \dots, 0) \in \{0, 1\}^{|\mathcal{X}|}$ with 1 in the kth element and 0s elsewhere. Let $\mathbf{y} \in \{0, 1\}^{|\mathcal{X}|}$ be obfuscated data. Then, for each element $i \in [|\mathcal{X}|]$, the UE mechanism outputs 1 with the following probabilities:

$$\Pr(\mathbf{y}[i] = 1|\mathbf{e}_k[i] = 1) = p, \quad \Pr(\mathbf{y}[i] = 1|\mathbf{e}_k[i] = 0) = q \tag{3}$$

where $p > q$. UE provides ε-LDP, where $\varepsilon = \ln([p(1-q)]/[(1-p)q])$.

RAPPOR [9] is a special case of UE where $p = (e^{\varepsilon/2}/e^{\varepsilon/2} + 1)$ and $q = (1/e^{\varepsilon/2} + 1)$. Wang et al. [23] proved that the UE mechanism minimizes the MSE when $p = (1/2)$ and $q = (1/e^{\varepsilon} + 1)$. They refer to UE with these parameters as optimal UE (OUE).

IDUE: As a randomized mechanism $\mathbf{Q}$ providing ID-LDP, we use the input-discriminative UE (IDUE) mechanism [12]. The IDUE mechanism is a modification of UE to provide ID-LDP.

For any $k \in [|\mathcal{X}|]$, the IDUE mechanism first maps x_k to the kth standard basis vector $\mathbf{e}_k \in \{0,1\}^{|\mathcal{X}|}$. Let $\mathbf{y} \in \{0,1\}^{|\mathcal{X}|}$ be obfuscated data. Then, for each element $i \in [|\mathcal{X}|]$, the IDUE mechanism outputs 1 with the following probabilities:

$$\Pr(\mathbf{y}[i] = 1 | \mathbf{e}_k[i] = 1) = a_i, \quad \Pr(\mathbf{y}[i] = 1 | \mathbf{e}_k[i] = 0) = b_i \tag{4}$$

where $a_i > b_i$ for any $i \in [|\mathcal{X}|]$. By (3) and (4), IDUE differs from UE in that it assigns different flip probabilities to different bits.

By (1), (4), and simple calculations, the following proposition holds (see [12] for the proof).

Proposition 1: The IDUE mechanism $\mathbf{Q}$ with

$$\frac{a_i\left(1-b_j\right)}{b_i\left(1-a_j\right)} \le e^{r\left(\varepsilon_{x_i}, \varepsilon_{x_j}\right)}$$

for any $i, j \in [|\mathcal{X}|]$ provides *$\mathcal{E}$-ID-LDP*.

Gu et al. [12] proposed an IDUE mechanism $\mathbf{Q}$ that minimizes the MSE. Assume that the input domain $\mathcal{X}$ is divided into $t \in \mathbb{N}$ subsets $\mathcal{X}_1, \ldots, \mathcal{X}_t$ according to privacy budgets; i.e., all input values have the same privacy budget within each subset. Let $m_i = |\mathcal{X}_i|$. Then, the optimization problem can be written as follows (see [12] for details):

$$\begin{aligned} \min_{a_i, b_i} \quad & \sum_{i=1}^{t} \frac{m_i b_i (1-b_i)}{(a_i - b_i)^2} + \max\left\{\frac{1 - a_i - b_i}{a_i - b_i}\right\} \\ \text{s.t.} \quad & \frac{a_i\left(1-b_j\right)}{b_i\left(1-a_j\right)} \le e^{r\left(\varepsilon_{x_i}, \varepsilon_{x_j}\right)} \quad (\forall i, j = 1, 2, \ldots, t) \\ & 0 < b_i < a_i < 1 \quad (\forall i, j = 1, 2, \ldots, t). \end{aligned} \tag{5}$$

The objective function represents the upper bound of the MSE. The constraints are imposed to satisfy ID-LDP.

The optimization problem in (5) is nonconvex. In our experiments, we used FindMinimum[3] in Mathematica as a solver for nonconvex optimization problems.

IV. Automatic Tuning of Privacy Budgets in Input-Discriminative LDP

ID-LDP provides fine-grained protection for input values with different sensitivity. A crucial issue in ID-LDP is that appropriate values of privacy budgets ($|\mathcal{X}|$ budgets in total) are unknown, as explained in Section I. Since the possible number of input values can be very large in IoT devices, it is also extremely difficult to manually set and check an appropriate privacy budget for each input value.

To address this issue, we propose algorithms to automatically determine privacy budgets in ID-LDP so that obfuscated data prevent reidentification, which is considered a major risk in GDPR [19]. We first introduce OneID-LDP as a new instance of ID-LDP and prove that OneID-LDP can be used to bound a reidentification risk. Then, we propose algorithms for automatically tuning privacy budgets in OneID-LDP to prevent reidentification.

Section IV-A describes the overview of our approach. Section IV-B introduces OneID-LDP. Section IV-C formalizes a reidentification risk. Section IV-D (resp., IV-E) shows the relationship between OneID-LDP (resp., MinID-LDP) and the reidentification risk. Section IV-F proposes our privacy budget tuning algorithms. The proofs of all statements in this section are given in Appendix A.

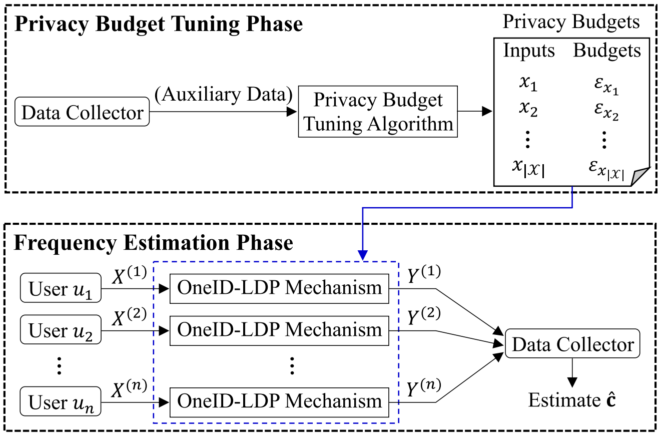


Fig. 1. Overview of our approach. The privacy budget tuning algorithm can optionally take auxiliary data as input.

A. Overview

Fig. 1 shows the overview of our approach. Our approach consists of two phases: 1) *privacy budget tuning phase* and 2) *frequency estimation phase*.

In the privacy budget tuning phase, a data collector calculates privacy budgets $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$ using a *privacy budget tuning algorithm* proposed in this article. This algorithm outputs privacy budgets $\mathcal{E}$ such that obfuscated data prevent reidentification. It can optionally take some auxiliary data as input. We propose one budget tuning algorithm without any auxiliary data and two budget tuning algorithms with auxiliary data. We explain their details in Section IV-F. After calculating privacy budgets $\mathcal{E}$, the data collector distributes $\mathcal{E}$ to each user.

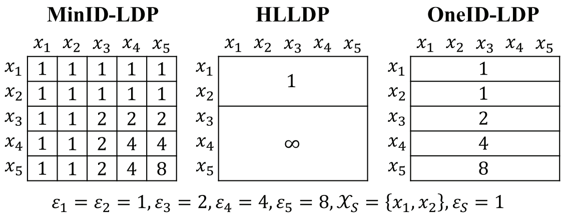
In the frequency estimation phase, each user $u_i \in \mathcal{U}$ uses a randomized mechanism providing OneID-LDP, which is introduced in Section IV-B. By using OneID-LDP as a privacy metric, we can strongly prevent reidentification, as explained in Sections IV-C and IV-D. Each user u_i obfuscates her personal data $X^{(i)}$ using OneID-LDP with privacy budgets $\mathcal{E}$ and sends obfuscated data $Y^{(i)}$ to the data collector. Finally, the data collector calculates an estimate $\hat{\mathbf{c}}$ of the frequency distribution $\mathbf{c}$ from the obfuscated data.

B. OneID-LDP

We now introduce OneID-LDP as a privacy metric. As described in Section III-C, MinID-LDP adopts the minimum of ε_x and $\varepsilon_{x'}$ as a privacy budget for a pair of x and x' (see Definition 3). In contrast, OneID-LDP uses only *one* privacy budget ε_x for this pair. Formally, it is defined as follows.

Definition 5 ($\mathcal{E}$-OneID-LDP): Let $\varepsilon_x \in \mathbb{R}_{\ge 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$. A randomized mechanism $\mathbf{Q}$ provides *$\mathcal{E}$-OneID-LDP* if and only if it provides $\mathcal{E}$-ID-LDP with $r(\varepsilon_x, \varepsilon_{x'}) = \varepsilon_x$.

[3] We also confirmed that FindMinimum provides higher utility than NMinimize, another solver for nonconvex optimization problems.



MinID-LDP

	x_1	x_2	x_3	x_4	x_5
x_1	1	1	1	1	1
x_2	1	1	1	1	1
x_3	1	1	2	2	2
x_4	1	1	2	4	4
x_5	1	1	2	4	8

HLLDP

	x_1	x_2	x_3	x_4	x_5
x_1	1				
x_2					
x_3	∞				
x_4					
x_5					

OneID-LDP

	x_1	x_2	x_3	x_4	x_5
x_1	1				
x_2	1				
x_3	2				
x_4	4				
x_5	8				

$\varepsilon_1 = \varepsilon_2 = 1, \varepsilon_3 = 2, \varepsilon_4 = 4, \varepsilon_5 = 8, \mathcal{X}_S = \{x_1, x_2\}, \varepsilon_S = 1$

Fig. 2. Example of pair budgets $r(\varepsilon_x, \varepsilon_{x'})$ in MinID-LDP, HLLDP, and OneID-LDP.

In other words, $\mathbf{Q}$ provides *$\mathcal{E}$-OneID-LDP* if and only if for any $x, x' \in \mathcal{X}$ and any $y \in \mathcal{Y}$

$$\mathbf{Q}(y|x) \le e^{\varepsilon_x}\mathbf{Q}(y|x'). \tag{6}$$

Fig. 2 shows an example of pair budgets $r(\varepsilon_x, \varepsilon_{x'})$ in MinID-LDP, HLLDP, and OneID-LDP when $\varepsilon_1 = \varepsilon_2 = 1$, $\varepsilon_3 = 2$, $\varepsilon_4 = 4$, $\varepsilon_5 = 8$, $\mathcal{X} = \{x_1, x_2\}$, and $\varepsilon_S = 1$. In this example, x_1 and x_2 are more sensitive than the others, and x_5 is the least sensitive. MinID-LDP adopts the minimum of ε_x and $\varepsilon_{x'}$ as a pair budget for x and x'. Consequently, it uses a small pair budget ($= 1$ or 2) for most pairs. Thus, MinID-LDP lacks utility, as shown in our experiments.

HLLDP is a special case of OneID-LDP where ε_x is set by (2). HLLDP sets the privacy budget of nonsensitive data (x_3, x_4, and x_5) to ∞. This is too drastic and leads to reidentification, as shown in our experiments.

In contrast, OneID-LDP provides more fine-grained protection for each input value and prevents reidentification attacks, as shown in Section IV-D. Moreover, OneID-LDP uses only one privacy budget ε_x for a pair of x and x'. Therefore, it uses large pair budgets for less sensitive data (x_3, x_4, and x_5). Consequently, OneID-LDP provides much higher utility than MinID-LDP while preventing reidentification.

It is well known that DP has basic properties, such as compositionality and immunity to post-processing [7], [15]. OneID-LDP also has these properties.

Proposition 2 (Sequential Composition): Let $\varepsilon_x^{(1)}, \varepsilon_x^{(2)} \in \mathbb{R}_{\ge 0}$ be privacy budgets for personal data $x \in \mathcal{X}$. Let $\mathcal{E}^{(1)} = \{\varepsilon_x^{(1)}\}_{x\in\mathcal{X}}$ and $\mathcal{E}^{(2)} = \{\varepsilon_x^{(2)}\}_{x\in\mathcal{X}}$. Let $\mathbf{Q}^{(1)}$ be a randomized mechanism and $y \in \mathcal{Y}$ be its output. If $\mathbf{Q}^{(1)}$ provides $\mathcal{E}^{(1)}$-OneID-LDP and $\mathbf{Q}^{(2)}(y)$ provides $\mathcal{E}^{(2)}$-OneID-LDP for any $y \in \mathcal{Y}$, then the sequential composition of $\mathbf{Q}^{(1)}$ and $\mathbf{Q}^{(2)}$ provides $\mathcal{E}$-OneID-LDP, where $\mathcal{E} = \{\varepsilon_x^{(1)} + \varepsilon_x^{(2)}\}_{x\in\mathcal{X}}$.

Proposition 3 (Post-Processing): Let $\varepsilon_x \in \mathbb{R}_{\ge 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x\in\mathcal{X}}$. Let λ be a randomized algorithm. If a randomized mechanism $\mathbf{Q}$ provides $\mathcal{E}$-OneID-LDP, then the composite function $\lambda \circ \mathbf{Q}$ provides $\mathcal{E}$-OneID-LDP.

For example, assume that a user obfuscates k (> 1) data using a mechanism providing $\mathcal{E}$-OneID-LDP, where $\mathcal{E} = \{\varepsilon_x\}_{x\in\mathcal{X}}$. Then by Proposition 2, we obtain $\mathcal{E}^*$-OneID-LDP in total, where $\mathcal{E}^* = \{k\varepsilon_x\}_{x\in\mathcal{X}}$. By Proposition 3, this privacy guarantee is immune to any post-processing algorithm run by the data collector.

C. Formalizing Reidentification Risk

Next, we formalize a reidentification risk. Let U be a random variable representing a user in $\mathcal{U}$. Let Y be a random variable representing obfuscated data of U. We assume that user U sends obfuscated data Y to a data collector and that Y is leaked to an adversary. Since each user sends a single datum, a prior distribution of U before obtaining Y is uniform for this adversary;[4] i.e., $\Pr(U = u_i) = (1/n)$ for any $u_i \in \mathcal{U}$. Assume that Y takes a value $y \in \mathcal{Y}$. The adversary attempts to determine whether U is $u_1, u_2, \ldots$, or u_n based on $Y = y$.

Let $\mathbf{p}_{U|Y=y}$ be the posterior distribution, whose element $\mathbf{p}_{U|Y=y}(u_i)$ represents the posterior probability that U is u_i; i.e., $\mathbf{p}_{U|Y=y}(u_i) = \Pr(U = u_i|Y = y)$. Using the posterior distribution, we can define the reidentification accuracy of the Bayes classifier. Specifically, let $\mathsf{Acc}_{U|Y=y}$ be the following quantity:

$$\mathsf{Acc}_{U|Y=y} = \max_{u_i\in\mathcal{U}} \mathbf{p}_{U|Y=y}(u_i).$$

$\mathsf{Acc}_{U|Y=y}$ is the reidentification accuracy of the Bayes classifier after observing Y. In other words, it is the highest possible reidentification accuracy. $\mathsf{Acc}_{U|Y=y}$ is a reidentification risk caused by sending $Y = y$.

D. Relationship Between OneID-LDP and Reidentification Accuracy

We prove that OneID-LDP can be used to upper bound the reidentification accuracy $\mathsf{Acc}_{U|Y=y}$ by a desired value.

Theorem 1: Let $\varepsilon_x \in \mathbb{R}_{\ge 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x\in\mathcal{X}}$. Let $\mathbf{Q}$ be a randomized mechanism providing $\mathcal{E}$-OneID-LDP, where

$$\varepsilon_x = \begin{cases} \log \frac{\gamma(n-\mathbf{c}(x))}{n-\gamma\mathbf{c}(x)}, & \left(\text{if } \mathbf{c}(x) < \frac{n}{\gamma}\right) \\ \infty, & (\text{otherwise}) \end{cases} \tag{7}$$

and $\gamma \in [1, n]$. Then for any $y \in \mathcal{Y}$ output by $\mathbf{Q}$

$$\mathsf{Acc}_{U|Y=y} \le \frac{\gamma}{n}. \tag{8}$$

Theorem 1 states that if we set privacy budgets $\mathcal{E} = \{\varepsilon_x\}_{x\in\mathcal{X}}$ by (7), then we can upper bound the reidentification accuracy by (γ/n) for any obfuscated data y, hence, *any user*. Note that even if the adversary randomly guesses U, the reidentification accuracy is $(1/n)$. By (7), this accuracy is achieved when $\gamma = 1$ and $\varepsilon_x = 0$, i.e., no utility.

A study in [22] proposed a privacy notion that upper bounds an *average* reidentification accuracy over all users. However, this average notion is weak because some users can be victims; e.g., even if the average reidentification accuracy is 1%, the adversary may reidentify 1% of all users with high confidence. In contrast, OneID-LDP with $\mathcal{E}$ in (7) upper bounds the reidentification accuracy for *every user*; i.e., the adversary cannot reidentify any user with high confidence. Thus, OneID-LDP is very strong in that there are no victims.

The value γ should be larger than 1 and much smaller than n to guarantee a small reidentification risk for every user with high utility. For example, we set $\gamma = 100 \ll n$ in our experiments. Then by (7), $\varepsilon_x \approx \log\gamma$ for an unpopular input value

[4] Note that the adversary may obtain obfuscated data of all users rather than a single user. For such scenarios, we assume a naive Bayes classifier that independently identifies each datum because it is highly scalable and accurate [52]. Then, the prior distribution of U is uniform for the adversary.

x whose frequency is $\mathbf{c}(x) \ll (n/\gamma)$. ε_x is much larger or ∞ for a popular input value x whose frequency is $\mathbf{c}(x) \approx (n/\gamma)$ or more. This means that for popular input values, we can strongly prevent reidentification with very little noise.

E. Relationship Between MinID-LDP and Reidentification Accuracy

We prove that MinID-LDP can also upper bound the reidentification accuracy $\mathsf{Acc}_{U|Y=y}$ by a desired value.

Proposition 4: Let $\varepsilon_x \in \mathbb{R}_{\geq 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$. Let $\mathbf{Q}$ be a randomized mechanism providing $\mathcal{E}$-MinID-LDP, where

$$\varepsilon_x = \begin{cases} \log \frac{\gamma(n-\mathbf{c}(x))}{n-\gamma\mathbf{c}(x)}, & \left(\text{if } \mathbf{c}(x) < \frac{n}{\gamma}\right) \\ \infty, & (\text{otherwise}) \end{cases}$$

and $\gamma \in [1, n]$. Then for any $y \in \mathcal{Y}$ output by $\mathbf{Q}$

$$\mathsf{Acc}_{U|Y=y} \leq \frac{\gamma}{n}.$$

By Theorem 1 and Proposition 4, the privacy budgets are the same between OneID-LDP and MinID-LDP. This means that our privacy budget tuning algorithms for OneID-LDP, which are proposed in Section IV-F, can also be used for determining privacy budgets in MinID-LDP to prevent reidentification.

F. Automatic Tuning of Privacy Budgets $\mathcal{E}$

In Section IV-D, we showed that we can upper bound the reidentification accuracy $\mathsf{Acc}_{U|Y=y}$ by using OneID-LDP with privacy budgets $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$ in (7). However, $\mathcal{E}$ in (7) includes the true frequency distribution $\mathbf{c}$. Unfortunately, the data collector cannot obtain $\mathbf{c}$ in advance, because the goal for the data collector is to estimate $\mathbf{c}$.

Therefore, we propose three privacy budget tuning algorithms, all of which do not use the true frequency distribution $\mathbf{c}$. Our three algorithms differ in the auxiliary data used for input. The first algorithm does not use any auxiliary data as input and determines the privacy budget ε_x based on the worst case value (i.e., smallest possible value) of $\mathbf{c}(x)$. We refer to this algorithm as a *worst case tuning algorithm*. The second algorithm assumes that the data collector knows that $\mathbf{c}(x)$ is larger than or equal to some value for some input values x. Then it determines ε_x by using the prior knowledge as auxiliary data. We refer to this algorithm as a *prior-based tuning algorithm*. Note that this prior knowledge is weak in that the data collector does not know the value of $\mathbf{c}(x)$ itself. The third algorithm uses obfuscated data of some users output by OneID-LDP mechanisms as auxiliary data. It estimates a confidence interval of $\mathbf{c}(x)$ from the obfuscated data and determines ε_x based on the confidence interval. We refer to this algorithm as a *confidence interval tuning algorithm*. As explained in Section IV-E, all of the three algorithms can be applied to both OneID-LDP and MinID-LDP.

Below, we explain these algorithms in detail.

Worst Case Tuning: The worst case tuning algorithm uses the fact that ε_x in (7) takes the smallest value when $\mathbf{c}(x) = 0$. Specifically, it outputs privacy budgets $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$, where

$$\varepsilon_x = \log \gamma. \tag{9}$$

Then, the reidentification accuracy $\mathsf{Acc}_{U|Y=y}$ is bounded by (γ/n) for any $y \in \mathcal{Y}$.

The worst case tuning algorithm does not use any auxiliary data as input. The next two algorithms provide higher utility than this algorithm by using auxiliary data.

Prior-Based Tuning: The prior-based tuning algorithm uses some weak prior knowledge about the frequency count $\mathbf{c}(x)$. Specifically, it assumes that the data collector knows $\mathbf{c}(x)$ is larger than or equal to some threshold for some input values x. This assumption is reasonable in many practical scenarios. For example, suppose that the data collector wants to estimate a population distribution in 47 prefectures of Japan from two million users ($n = 2 \times 10^6$). It is well known that more than 11% of people live in Tokyo. Thus, the data collector would know that $\mathbf{c}(x) \geq 10^5$ for Tokyo. Note that the data collector does not know the exact value of $\mathbf{c}(x)$ in Tokyo. We can use OneID-LDP mechanisms with this prior knowledge to accurately estimate the exact value of $\mathbf{c}(x)$ in Tokyo.

Formally, let $\tilde{\mathcal{X}} \subseteq \mathcal{X}$ be the set of personal data for which the data collector has prior knowledge. The data collector knows that $\mathbf{c}(x)$ is larger than or equal to a threshold $\lambda(x) \in \mathbb{Z}_{\geq 0}$ for $x \in \tilde{\mathcal{X}}$. Then, the prior-based tuning algorithm assigns $\lambda(x)$ to $\mathbf{c}(x)$ for $x \in \tilde{\mathcal{X}}$ and 0 to $\mathbf{c}(x)$ for $x \notin \tilde{\mathcal{X}}$ in (7).

That is, the prior-based tuning algorithm takes $\lambda(x)$ for $x \in \tilde{\mathcal{X}}$ as input and outputs privacy budgets $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$, where

$$\varepsilon_x = \begin{cases} \log \frac{\gamma(n-\lambda(x))}{n-\gamma\lambda(x)}, & \left(\text{if } x \in \tilde{\mathcal{X}} \text{ and } \lambda(x) < \frac{n}{\gamma}\right) \\ \infty, & \left(\text{if } x \in \tilde{\mathcal{X}} \text{ and } \lambda(x) \geq \frac{n}{\gamma}\right) \\ \log \gamma, & \left(\text{if } x \notin \tilde{\mathcal{X}}\right). \end{cases} \tag{10}$$

Note that $\mathbf{c}(x) \geq \lambda(x)$ for $x \in \tilde{\mathcal{X}}$ and $\mathbf{c}(x) \geq 0$ for $x \notin \tilde{\mathcal{X}}$. Since ε_x in (7) is monotonically increasing with respect to $\mathbf{c}(x)$, $\mathsf{Acc}_{U|Y=y}$ is bounded by (γ/n) for any $y \in \mathcal{Y}$.

Confidence Interval Tuning: The prior-based tuning algorithm assumes weak prior knowledge about the frequency count $\mathbf{c}(x)$. Although this is reasonable in many practical scenarios, as explained above, it can happen that the data collector has no prior knowledge about $\mathbf{c}(x)$ in some cases. For example, the data collector may not have any prior about $\mathbf{c}(x)$ for health conditions collected from new wearable devices.

For this scenario, we propose the confidence interval tuning algorithm to estimate $\mathbf{c}$ more accurately than the worst case tuning algorithm. In the confidence interval tuning, we divide users into two groups: 1) *worst case group* and 2) *confidence interval group*. First, we use the worst case tuning algorithm for users in the worst case group and collect their obfuscated data providing OneID-LDP. Then, we estimate the confidence interval of $\mathbf{c}(x)$ from the obfuscated data. Based on the confidence interval, we determine privacy budgets $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$ for users in the confidence interval group. Finally, we collect their obfuscated data providing OneID-LDP and calculate an estimate $\hat{\mathbf{c}}$. All users are protected by OneID-LDP.

Formally, let $\mathcal{U}_0 \subseteq \mathcal{U}$ be the worst case group. Without loss of generality, we assume that the worst case group is $\mathcal{U}_0 = \{u_1, \ldots, u_{n_0}\}$, where $n_0 \in [n]$. Each user in the worst case group $\mathcal{U}_0$ obfuscates her personal data using an IDUE mechanism (described in Section III-D) to provide OneID-LDP with

Algorithm 1 Confidence Interval Tuning

```
Input: $\mathbf{Y}_0 = \{Y^{(1)}, \cdots, Y^{(n_0)}\}$, $\alpha \in \mathbb{R}_{\geq 0}$
Output: $\mathcal{E} = \{\varepsilon_{x_i} | 1 \leq i \leq |\mathcal{X}|\}$
 1: $z \leftarrow$ ZValue($\alpha$)
 2: for $i = 1$ to $|\mathcal{X}|$ do
 3:     $t_i \leftarrow$ FrequencyCount($\mathbf{Y}_0, i$)
 4:     $\mathbf{r}(x_i) \leftarrow \frac{t_i + \frac{1}{2}z^2}{n+z^2} - \frac{z}{n+z^2}\sqrt{\frac{t_i(n-t_i)}{n} + \frac{z^2}{4}}$
 5:     $\mathbf{c}(x_i) \leftarrow \max\{\frac{n}{a_i - b_i}(\mathbf{r}(x_i) - b_i), 0\}$
 6:     if $\mathbf{c}(x_i) < \frac{n}{\gamma}$ then
 7:         $\varepsilon_{x_i} \leftarrow \log \frac{\gamma(n - \mathbf{c}(x))}{n - \gamma \mathbf{c}(x)}$
 8:     else
 9:         $\varepsilon_{x_i} \leftarrow \infty$
10:     end if
11: end for
12: return $\mathcal{E} = \{\varepsilon_{x_i} | 1 \leq i \leq |\mathcal{X}|\}$
```

$\mathcal{E}$ in (9), i.e., worst case tuning. Then, she sends her obfuscated data to the data collector. Let $\mathbf{Y}_0 = \{Y^{(1)}, \ldots, Y^{(n_0)}\}$ be the set of obfuscated data of $\mathcal{U}_0$. The data collector estimates a confidence interval of $\mathbf{c}(x)$ from $\mathbf{Y}_0$ and determines $\mathcal{E}$ based on the interval.

Algorithm 1 shows our confidence interval tuning algorithm. Assume that the input domain is $\mathcal{X} = \{x_1, \ldots, x_{|\mathcal{X}|}\}$ without loss of generality. For $i \in [|\mathcal{X}|]$, the data collector estimates a confidence interval of $\mathbf{c}(x_i)$ and sets ε_{x_i} based on the interval (lines 2–11). Recall that the output range of IDUE is $\mathcal{Y} = \{0, 1\}^{|\mathcal{X}|}$. For $i \in [|\mathcal{X}|]$, let $t_i \in \mathbb{Z}_{\geq 0}$ be the number of "1"s in the ith bit of output data in $\mathbf{Y}_0$ (output of FrequencyCount($\mathbf{Y}_0, i$) in line 3). The relative frequency of "1" (resp., "0") in the ith bit of input data is $([\mathbf{c}(x_i)]/n)$ (resp., $1 - ([\mathbf{c}(x_i)]/n)$). Thus, by (4), the probability that the ith bit of output data is "1" can be written as: $([\mathbf{c}(x_i)]/n)a_i + (1 - ([\mathbf{c}(x_i)]/n))b_i = (a_i - b_i)([\mathbf{c}(x_i)]/n) + b_i$.

Let $\mathbf{r}(x_i)$ be the following quantity:

$$\mathbf{r}(x_i) = (a_i - b_i)\frac{\mathbf{c}(x_i)}{n} + b_i. \tag{11}$$

$\mathbf{r}(x_i)$ is the probability that the ith bit of output data is "1." Thus, we can assume that the number t_i of "1"s in the ith bit of output data is generated from the Binomial distribution $B(n, \mathbf{r}(x_i))$ with success probability $\mathbf{r}(x_i)$

$$t_i \sim B(n, \mathbf{r}(x_i)). \tag{12}$$

A confidence interval for $\mathbf{r}(x_i)$ is known as the *binomial proportion confidence interval* [53], [54], [55]. It can be estimated from t_i and n using estimators, such as the Normal approximation interval and the Wilson score interval.

We estimate the confidence interval of $\mathbf{r}(x_i)$ from t_i and n. Here, we use the Wilson score interval because it is accurate [53], [54], [55]. Specifically, let $z \in \mathbb{R}_{\geq 0}$ be the $1 - (\alpha/2)$ quantile of a standard normal distribution $\mathcal{N}(0, 1)$ corresponding to the significance level α (output of ZValue(α) in line 1). For example, for a 95% (resp., 99%) confidence interval, $\alpha = 0.05$ and $z = 1.96$ (resp., 2.576). Then, the Wilson score interval of $\mathbf{r}(x_i)$ is given by

$$\mathbf{r}(x_i) = \frac{t_i + \frac{1}{2}z^2}{n + z^2} \pm \frac{z}{n + z^2}\sqrt{\frac{t_i(n - t_i)}{n} + \frac{z^2}{4}}. \tag{13}$$

By (11), we can calculate the confidence interval of $\mathbf{c}(x_i)$ corresponding to $\mathbf{r}(x_i)$ in (13) (if $\mathbf{c}(x_i)$ becomes negative, we set $\mathbf{c}(x_i) = 0$). Since ε_x in (7) is a nondecreasing function of $\mathbf{c}(x)$, we adopt the minimum values of $\mathbf{r}(x_i)$ and $\mathbf{c}(x_i)$ in the intervals (lines 4 and 5). In other words, we consider the worst case about $\mathbf{c}(x_i)$ in the confidence interval. Then, we set $\mathcal{E} = \{\varepsilon_{x_i} | 1 \leq i \leq |\mathcal{X}|\}$ by (7) (lines 6–10).

Finally, each user in the confidence interval group $\mathcal{U} \setminus \mathcal{U}_0$ obfuscates her personal data using an IDUE mechanism to provide OneID-LDP with $\mathcal{E}$ output by the confidence interval tuning algorithm (Algorithm 1). The data collector estimates $\mathbf{c}$ from obfuscated data $\mathbf{Y}$. Note that the worst case group and confidence interval group use different randomized mechanisms. To deal with this difference, we calculate an empirical estimate [12] for each group and then calculate an estimate $\hat{\mathbf{c}}$ of $\mathbf{c}$ by the inverse-variance weighting [56] of the two empirical estimates.

Significance Level: The utility and privacy of our confidence interval tuning depend on the significance level α. If we set the significance level α to $\alpha = 0$, then $z = \infty$ and the estimated minimum value of $\mathbf{c}(x_i)$ becomes 0. Thus, the confidence interval tuning with $\alpha = 0$ is identical to the worst case tuning. As α is increased from 0, the privacy budgets $\mathcal{E}$ become larger and the utility is increased. However, the true frequency count $\mathbf{c}(x_i)$ can be smaller than the estimated minimum value with probability $(\alpha/2)$. Thus, the reidentification accuracy $\mathsf{Acc}_{U|Y=y}$ may exceed (γ/n) when α is too large.

In our experiments, we set $\alpha = 0.05$ and show that $\mathsf{Acc}_{U|Y=y}$ does not exceed (γ/n) in this case.

Which Tuning Algorithm to Use? We have so far proposed three privacy budget tuning algorithms. Here, we provide a guideline for which algorithm to use in practice.

As we will show in our experiments, an appropriate tuning algorithm depends on the task of the data collector and the prior knowledge about the frequency count $\mathbf{c}(x)$. In some tasks, frequency counts of *popular* input values [e.g., $\mathbf{c}(x) \geq (n/\gamma)$] are, especially, important; e.g., they are used for finding popular POIs [2] and automatic labeling of POIs, such as offices and schools [57]. For popular input values, the worst case tuning provides the lowest utility, and the prior-based tuning provides the highest utility. Thus, if we have some prior knowledge about $\mathbf{c}(x)$, we should use the prior-based tuning algorithm. Otherwise, the confidence interval tuning could be the best choice.

However, for *unpopular* input values [e.g., $\mathbf{c}(x) \ll (n/\gamma)$], our three tuning algorithms provide almost the same utility. Thus, if we want to estimate frequency counts of unpopular input values, the worst case tuning would be sufficient.

V. Experimental Evaluation

In this section, we show through experiments that our algorithms provide much higher utility than LDP mechanisms while preventing reidentification. We also show that

TABLE IV
NUMBERS OF USERS AND INPUT VALUES IN EACH DATA SET

	IST	NYK	TKY	SP	KL	JK	Local	ADL	RFID
#users n	330159	88064	51719	67697	80160	151315	164860	741	75128
#input values $\lvert\mathcal{X}\rvert$	70557	25483	21008	25188	24493	62373	11	10	4
#sensitive input values	9459	946	212	2532	2461	11261	2	3	1

existing ID-LDP mechanisms (i.e., MinID-LDP and HL-LDP mechanisms) lack either utility or privacy.

Section V-A explains our experimental setup. Section V-B reports our experimental results.

A. Experimental Setup

Data Set: We conducted experiments using the following four real data sets.

1) *Foursquare Data Set:* The Foursquare data set (Global-scale Check-in Data Set with User Social Networks) [58] is a large-scale location data set. It includes 90 048 627 check-ins all over the world, each of which is associated with a POI ID and venue category (e.g., hospital, restaurant, park, and university). Following [58], we selected six cities with numerous check-ins and with cultural diversities: Istanbul (denoted by IST), New York (NYK), Tokyo (TKY), San Paulo (SP), Kuala Lumpur (KL), and Jakarta (JK). We extracted one check-in from each user.
2) *Localization Data Set:* The Localization data set [3] (denoted by Local) is a person activity data set collected using wearable sensors. It includes 164 860 records, each of which has an activity value, such as walking, falling, lying, and on all fours (11 values in total).
3) *ADL Data Set:* The activities of daily living (ADL) data set [59] (denoted by ADL) is a person activity data set collected using a wireless sensor network. It includes 741 records, each of which has an activity value, such as toileting, sleeping, showering, and lunch (10 values in total).
4) *RFID Data Set:* The RFID-based activity recognition data set [60] (denoted by RFID) is a person activity data set collected from older people using RFID reader antennas around rooms. It includes 75 128 records, each of which has an activity value, such as sitting on a bed, lying on a bed, and ambulating (4 values in total).

In Local, ADL, and RFID, we assumed that each record is from a different user. In the Foursquare data set, we assumed that input values (POIs) with "home" or "hospital" categories are sensitive. In the person activity data sets, we assumed sleeping (or lying/lying down), toileting, and showering as sensitive because they reveal detailed life patterns. We set the privacy budgets for these sensitive input values to 1 [15] to strongly protect them, as described in Section I.

Table IV shows the number of users, input values, and sensitive input values in each data set.

Randomized Mechanisms: Using the four data sets, we evaluated the following randomized mechanisms.

1) *RAPPOR:* Google's RAPPOR [9] described in Section III-D. It provides ε-LDP.
2) *OUE:* The optimal unary encoding (OUE) mechanism [23] described in Section III-D. It provides ε-LDP.
3) *MinID-LDP Mechanism:* A randomized mechanism providing $\mathcal{E}$-MinID-LDP. To provide MinID-LDP, we used the optimal IDUE mechanism in Section III-D.
4) *HLLDP Mechanism:* A randomized mechanism providing $(\mathcal{X}_S, \varepsilon_S)$-HLLDP. Specifically, we used the utility-optimized RAPPOR [13] as an HLLDP mechanism.
5) *OneID-LDP Mechanism:* Our $\mathcal{E}$-OneID-LDP mechanism in Section IV-B. To provide OneID-LDP, we used the optimal IDUE mechanism.

Parameters: We set the privacy budgets for sensitive input values to 1, as explained above. Since LDP regards all input values as equally sensitive, we set $\varepsilon = 1$ for RAPPOR and OUE. For HLLDP, we set the privacy budget to $\varepsilon_S = 1$ for the sensitive input values $\mathcal{X}_S$ and ∞ for the remaining input values. For the MinID-LDP and OneID-LDP mechanisms, we used our privacy budget tuning algorithms to determine the privacy budgets $\mathcal{E}$.

Our privacy budget tuning algorithms have three parameters: γ, α, and n_0 (α and n_0 are used only in the confidence interval tuning algorithm). In our experiments, we set $\gamma = 10$ or 100; i.e., we set $\mathcal{E}$ so that the reidentification accuracy is smaller than $(10/n)$ or $(100/n)$ (see Theorem 1). In the prior-based tuning algorithm, we assumed that the data collector knows popular personal data x whose frequency $\mathbf{c}(x)$ is larger than or equal to (n/γ). In other words, we used the set of the popular personal data as $\tilde{\mathcal{X}}$ and set $\lambda(x) = (n/\gamma)$ for $x \in \tilde{\mathcal{X}}$. In the confidence interval tuning algorithm, we set the significance level α to $\alpha = 0.01$ or 0.05 and assumed that 10% or 50% of users are in the worst case group; i.e., $n_0 = 0.1n$ or $0.5n$.

We set $\gamma = 100$, $\alpha = 0.05$, and $n_0 = 0.1n$ as default values. Then, we changed each of γ, α, and n_0 while fixing the other two to see how each parameter affects the performance.

Utility and Privacy: We evaluated the utility and privacy of the randomized mechanisms.

For utility loss, we evaluated the MAE and MSE over all input values, as described in Section III-B. We also evaluated the MAE and MSE over popular input values x whose frequency counts $\mathbf{c}(x)$ are larger than or equal to $(n/100)$.

For privacy, we considered the following reidentification attack. In our experiments, the input domain is $\mathcal{X} = \{x_1, \ldots, x_{|\mathcal{X}|}\}$ and the output range is $\mathcal{Y} = \{0, 1\}^{|\mathcal{X}|}$. Given obfuscated data $y \in \mathcal{Y}$, the adversary extracts indices whose corresponding values in y are 1. Then, the adversary chooses an index i whose privacy budget ε_{x_i} is the largest among the extracted indices. Finally, the adversary outputs a user who has x_i as a reidentification result (if multiple users have personal data with the largest privacy budget, then the adversary randomly outputs one user from them).

Note that this adversary knows the values of privacy budgets and each user's personal data x_i, i.e., maximum-knowledge attacker [61], [62]. The maximum-knowledge attacker model is useful for evaluating reidentification risks when we assume

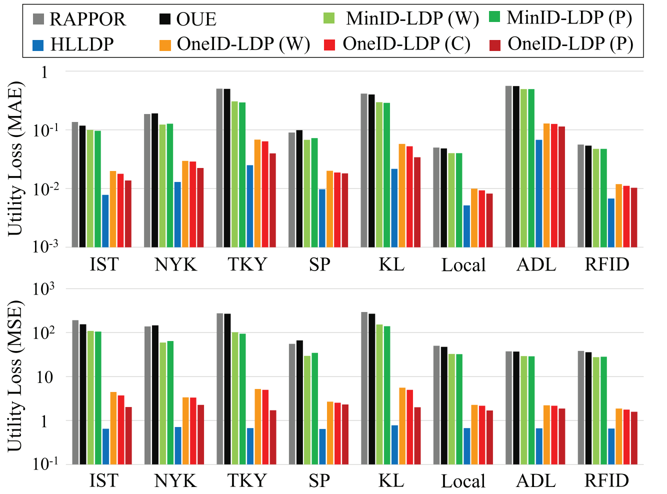


Fig. 3. MAE and MSE over popular input values where $\mathbf{c}(x) \geq n/100$ (W: worst case tuning, C: confidence interval tuning, P: prior-based tuning, $\gamma = 100$, $\alpha = 0.05$, and $n_0 = 0.1n$). Smaller values mean higher utility. In JK, there is no such popular input value. Note that HLLDP is vulnerable to reidentification attacks, as shown in Figs. 5 and 6.

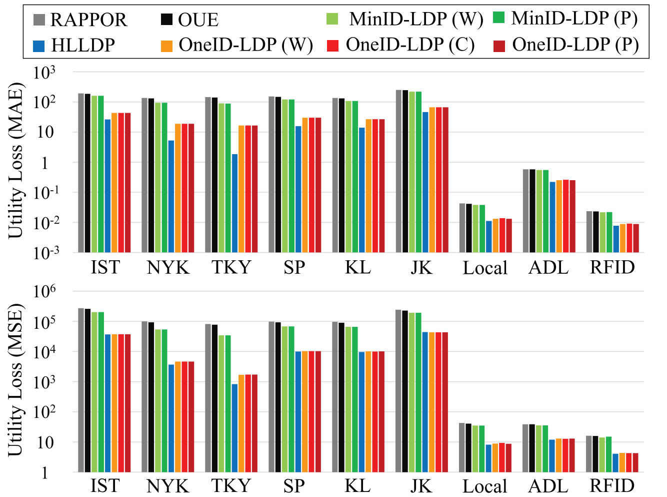


Fig. 4. MAE and MSE over all input values (W: worst case tuning, C: confidence interval tuning, P: prior-based tuning, $\gamma = 100$, $\alpha = 0.05$, and $n_0 = 0.1n$). Smaller values mean higher utility. Note that HLLDP is vulnerable to reidentification attacks, as shown in Figs. 5 and 6.

a worst case scenario about the adversary's background knowledge. It also poses a threat in some practical situations. For example, if a user sends some additional information (e.g., time and health condition) along with x_i (e.g., location) from her wearable device, the adversary can link the additional information to the user by this reidentification attack. The linked information may also be used for reidentifying other databases or making a user profile, as described in Section I.

We implemented the above reidentification attack and evaluated a reidentification rate, which is the proportion of correctly identified data. For both utility and privacy, we ran a randomized mechanism 1000 times and evaluated the average performance.

B. Experimental Results

Utility: Figs. 3 and 4 show the MAE/MSE over popular input values and all input values, respectively (W, C, and P in the parentheses represent the worst case tuning, confidence interval tuning, and prior-based tuning, respectively). Here, we set $\gamma = 100$, $\alpha = 0.05$, and $n_0 = 0.1n$ (later, we will change the values of γ, α, and n_0). In JK, there is no popular input values such that $\mathbf{c}(x) \geq (n/100)$. Thus, we do not show the results for JK in Fig. 3.

Figs. 3 and 4 show that LDP mechanisms (RAPPOR and OUE) provide poor utility. This is because LDP regards all input values as equally sensitive. MinID-LDP provides utility similar to LDP, and the prior-based tuning does not improve the utility of MinID-LDP. This is because MinID-LDP uses a small pair budget when either of the two input values is sensitive. In other words, it still overprotects personal data. Figs. 3 and 4 also show that HLLDP provides the highest utility. However, it comes at the expense of privacy—later, we will show that HLLDP is vulnerable to reidentification attacks and *cannot* be used for our purpose of privacy protection.

Except for the insecure HLLDP, our OneID-LDP mechanisms provide the best performance. They outperform LDP and MinID-LDP mechanisms by one or two orders of magnitude. For popular input values, OneID-LDP (C) outperforms OneID-LDP (W), and OneID-LDP (P) provides the highest utility (see Fig. 3). For example, the MSEs of OneID-LDP (W), OneID-LDP (C), and OneID-LDP (P) in IST were 4.43, 3.72, and 2.04, respectively. In contrast, for all input values, all of our three OneID-LDP mechanisms provide almost the same utility (see Fig. 4). This is because most of the input values are unpopular ($\mathbf{c}(x) \ll (n/\gamma)$) and $\varepsilon_x \approx \log \gamma$ for these input values in all of our three OneID-LDP mechanisms. In other words, the worst case tuning is sufficient for estimating the frequency counts of unpopular input values.

Thus, an appropriate tuning method depends on the task and the prior knowledge about $\mathbf{c}(x)$. If we want to accurately estimate popular input values and have some prior knowledge about $\mathbf{c}(x)$, then we should use the prior-based tuning. If we want to estimate popular input values without any prior, then we could use the confidence interval tuning. Otherwise, the worst case tuning would be sufficient.

Privacy: Next, we evaluated the reidentification risk. Fig. 5 shows the reidentification rate for all users. when $\gamma = 100$, $\alpha = 0.05$, and $n_0 = 0.1n$ (later, we will change γ, α, and n_0). We show the results for HLLDP and our three OneID-LDP mechanisms (W, C, and P). We do not show the results for RAPPOR, OUE, and MinID-LDP, because they lack utility as shown in Figs. 3 and 4.

Fig. 5 shows that all of our three OneID-LDP mechanisms (W, C, and P) keep the reidentification rate smaller than the required value ($= [\gamma/n]$). This is because our privacy budget tuning algorithms determine privacy budgets in OneID-LDP so that the reidentification accuracy is bounded by (γ/n), as described in Section IV-F. In contrast, the reidentification rate of HLLDP is much higher than the required value in the Foursquare data set. This is because HLLDP assigns $\varepsilon_x = \infty$ for nonsensitive data and reveals the corresponding input values. In Local, ADL, and RFID, the number $|\mathcal{X}|$ of input values is very small, as shown in Table IV. Thus, many users have the same input value, and reidentification is difficult in these data sets. However, $|\mathcal{X}|$ is very large in the Foursquare data set,

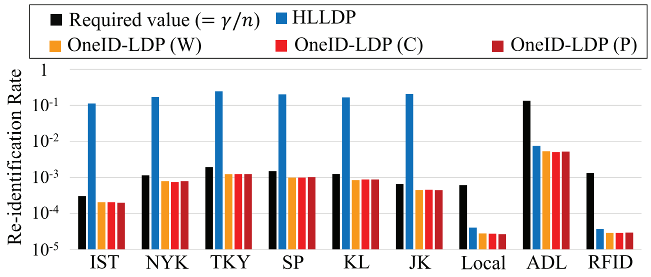


Fig. 5. Reidentification rate for all users (W: worst case tuning, C: confidence interval tuning, P: prior-based tuning, $\gamma = 100$, $\alpha = 0.05$, and $n_0 = 0.1n$).

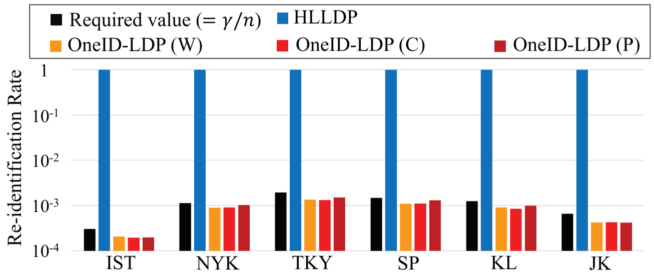


Fig. 6. Reidentification rate for outliers in the Foursquare data set (W: worst case tuning, C: confidence interval tuning, P: prior-based tuning, $\gamma = 100$, $\alpha = 0.05$, and $n_0 = 0.1n$). Outliers have a unique input value and an output value with at least one "1" in nonsensitive bits.

and consequently, many users have a "unique" input value; i.e., many input values are associated with only one user. Therefore, HLLDP is vulnerable to the reidentification attack in the Foursquare data set.

To show the vulnerability of HLLDP more comprehensively, we also evaluated the reidentification rate for "outliers" who have a unique input value and an output value with at least one "1" in nonsensitive bits. Fig. 6 shows the results in the Foursquare data set. We observe that the reidentification rate of HLLDP is 100%. This is because, in the HLLDP mechanism in [13], every output value with at least one "1" in nonsensitive bits reveals the corresponding input value. These output data are called *invertible data* in [13]. Since the invertible data reveal the corresponding input values, HLLDP allows the adversary to perfectly reidentify the outliers. Thus, HLLDP cannot be used to prevent reidentification.

In contrast, our three OneID-LDP mechanisms keep the reidentification rate smaller than the required value (= $[\gamma/n]$) even for the outliers. This is because OneID-LDP upper bounds the reidentification accuracy by (γ/n) for any obfuscated data $y \in \mathcal{Y}$, hence, any user (Theorem 1).

Note that MinID-LDP also upper bounds the reidentification accuracy by (γ/n) because MinID-LDP uses smaller privacy budgets than OneID-LDP. However, it comes at the cost of utility, as shown in Figs. 3 and 4.

Effects of Parameters: We also examined how the parameters γ, α, and n_0 in our privacy budget tuning algorithms affect the utility and privacy. Fig. 7 shows the MAE/MSE over all input values[5] when we set $\gamma = 100$ or 10. In addition, Figs. 8 and 9 show the reidentification rate for all users and outliers, respectively, when we set $\gamma = 100$ or 10.

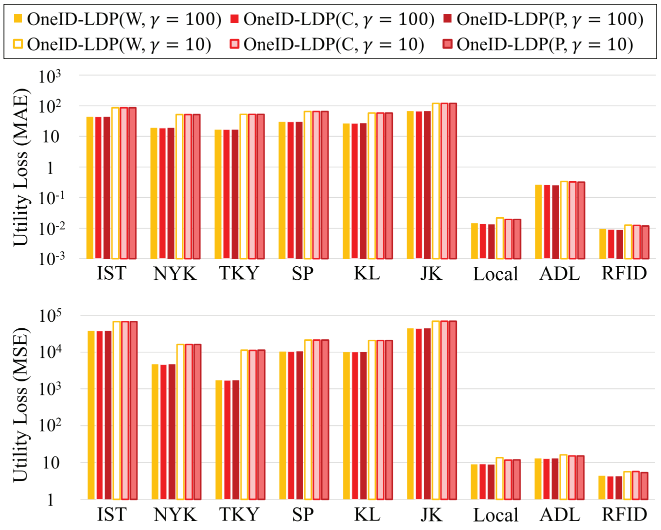


Fig. 7. Effect of the parameter γ on the MAE and MSE over all input values (W: worst case tuning, C: confidence interval tuning, P: prior-based tuning, $\alpha = 0.05$, and $n_0 = 0.1n$). Smaller values mean higher utility.

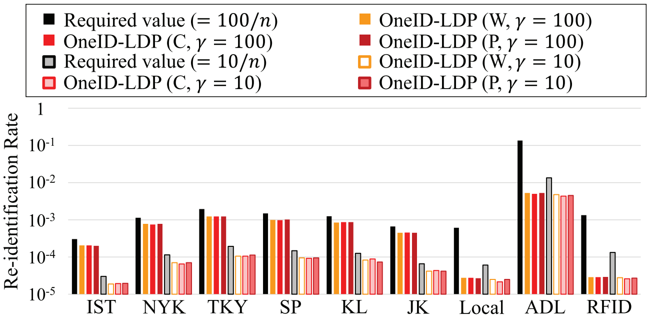


Fig. 8. Effect of the parameter γ on the reidentification rate for all users (W: worst case tuning, C: confidence interval tuning, P: prior-based tuning, $\alpha = 0.05$, and $n_0 = 0.1n$).

Figs. 7–9 show that γ controls the privacy–utility tradeoff—as γ decreases from 100 to 10, the privacy is improved at the cost of utility. Figs. 8 and 9 also show that our OneID-LDP mechanisms keep the reidentification rate smaller than the required value (= $[\gamma/n]$), irrespective of the value of γ. This result demonstrates that our privacy budget tuning algorithms successfully determine the privacy budgets so that obfuscated data prevent reidentification, as desired.

Finally, we examined the effect of the other two parameters α and n_0 in our confidence interval tuning algorithm. Figs. 10–12 show the MAE/MSE over popular input values, the MAE/MSE over all input values, and the reidentification rate, respectively, when we change α and n_0.

Fig. 10 shows that as the significance level α decreases from 0.05 to 0.01, the utility becomes worse, especially, in IST and KL. This result is expected, as the privacy budgets decrease with decrease in α. Fig. 12 shows that the privacy is slightly improved with decrease in α. However, our OneID-LDP mechanisms keep the reidentification rate smaller than the required value even when $\alpha = 0.05$, as shown in Figs. 8 and 9. Thus,

[5] We do not evaluate the MAE/MSE over popular input values, because there is no popular input value such that $\mathbf{c}(x) \geq (n/10)$ in the Fourqaure data set.

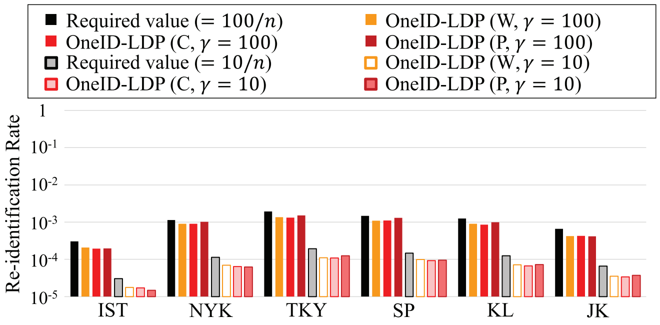


Fig. 9. Effect of the parameter γ on the reidentification rate for outliers (W: worst case tuning, C: confidence interval tuning, P: prior-based tuning, $\alpha = 0.05$, and $n_0 = 0.1n$).

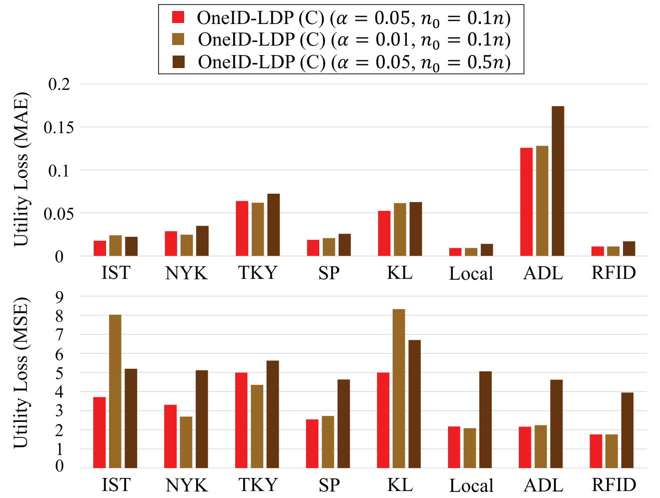


Fig. 10. Effects of the parameters α and n_0 on the MAE and MSE over popular input values (C: confidence interval tuning and $\gamma = 100$). Smaller values mean higher utility.

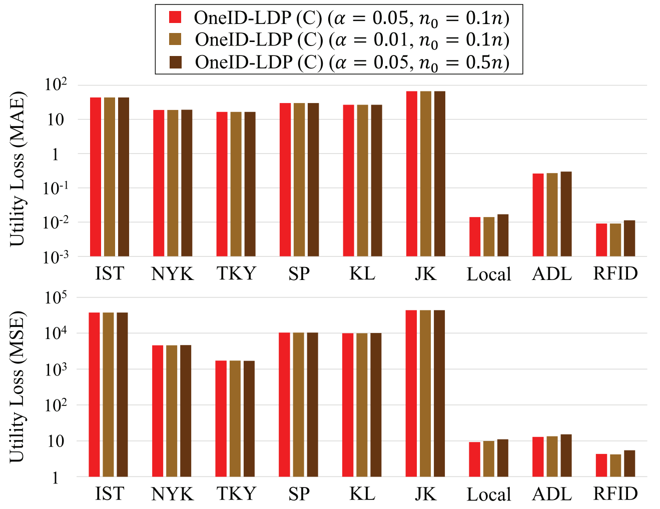


Fig. 11. Effects of the parameters α and n_0 on the MAE and MSE over all input values (C: confidence interval tuning and $\gamma = 100$). Smaller values mean higher utility.

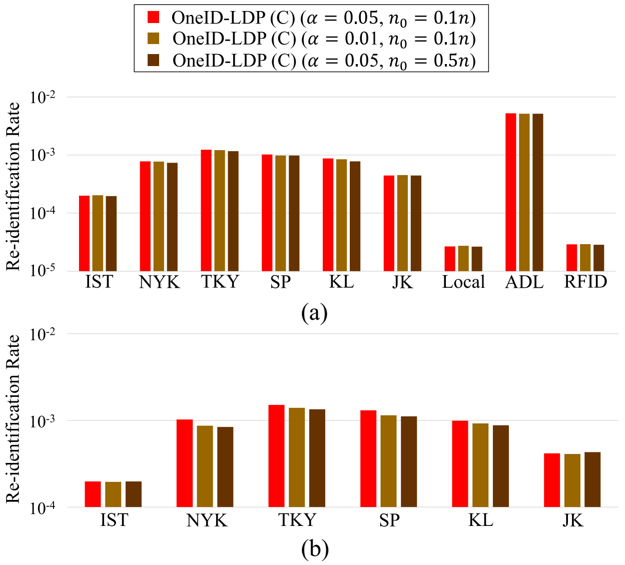


Fig. 12. Effects of the parameters α and n_0 on the reidentification rate (C: confidence interval tuning and $\gamma = 100$). (a) All users. (b) Outliers.

$\alpha = 0.05$ is sufficient for the purpose of privacy protection in our experiments.

Figs. 10–12 also show that as the number n_0 of users in the worst case group increases from $0.1n$ to $0.5n$, the utility becomes worse, and the privacy is slightly improved. This is because the users in the worst case groups have smaller privacy budgets. Note that when $n_0 = n$, our confidence interval tuning is equivalent to the worst case tuning. Our confidence interval tuning provides higher utility when there are a lot of users in the confidence interval group.

Summary: In summary, our experimental results show that the existing instances of ID-LDP lack either utility or privacy. Specifically, Min-IDP still overprotects personal data, and, therefore, its utility gain over LDP (i.e., RAPPOR and OUE) is limited. HLLDP assigns $\varepsilon_x = \infty$ for nonsensitive data and, therefore, is vulnerable to the reidentification attack.

In contrast, our OneID-LDP mechanisms with our privacy budget tuning algorithms provide much higher utility than LDP and Min-LDP mechanisms while keeping the reidentification accuracy smaller than the required value. Thus, our mechanisms can be used for accurate analysis of personal data collected from IoT devices while strongly preventing reidentification, which is considered to be a major risk in GDPR.

One limitation of our proposed methods is that our confidence interval tuning algorithm does not theoretically upper bound the reidentification accuracy. As described in Section IV-F (“Significance Level”), the reidentification accuracy may exceed the required value (γ/n) when the significance level α is too large. If we want to theoretically upper bound the reidentification accuracy, we should use the worst case tuning algorithm or the prior-based tuning algorithm.

VI. Conclusion

We proposed three privacy budget tuning algorithms for ID-LDP to provide high utility while preventing reidentification. We also proposed OneID-LDP as a new instance of ID-LDP and proved that it upper bounds the reidentification accuracy for every user. Through experiments using four real data sets, we showed that existing ID-LDP mechanisms lack either utility or privacy. Then, we showed that our OneID-LDP mechanisms with our privacy budget tuning algorithms provide much higher utility than LDP mechanisms while keeping the reidentification accuracy smaller than the required value.

In this article, we focused on frequency estimation of personal data such as locations and person activity data as a task of the data collector. As future work, we would like to develop ID-LDP mechanisms and privacy budget tuning algorithms for more complicated tasks, such as item recommendation [63] and subgraph counting in a social graph [34].

Appendix A
Proof of Theorem 1

Recall that U and Y are random variables representing a user and obfuscated data of U, respectively. Let X be a random variable representing personal data of U. Personal data X is uniquely determined given U. Let x be the personal data of u_i. Then, $\mathbf{c}(x) \geq 1$ and the posterior probability $\mathbf{p}_{U|Y=y}(u_i)$ can be written as

$$\begin{aligned}
&\mathbf{p}_{U|Y=y}(u_i)\\
&\quad= \Pr(U = u_i|Y = y)\\
&\quad= \Pr(U = u_i|X = x)\Pr(X = x|Y = y)\\
&\quad= \frac{1}{\mathbf{c}(x)}\Pr(X = x|Y = y)\\
&\qquad\text{(as there are } \mathbf{c}(x) \text{ users whose input value is } x\text{).} \qquad (14)
\end{aligned}$$

Below, we write $\Pr(x|y)$ as a shorthand of $\Pr(X = x|Y = y)$. By Bayes' theorem, we have

$$\begin{aligned}
&\Pr(x|y)\\
&\quad= \frac{\Pr(y|x)\Pr(x)}{\sum_{x' \in \mathcal{X}} \Pr(y|x')\Pr(x')}\\
&\quad= \frac{\Pr(y|x)\Pr(x)}{\Pr(y|x)\Pr(x) + \sum_{x' \neq x} \Pr(y|x')\Pr(x')}\\
&\quad\leq \frac{\Pr(y|x)\Pr(x)}{\Pr(y|x)\Pr(x) + \sum_{x' \neq x} e^{-\varepsilon_x}\Pr(y|x)\Pr(x')} \quad \text{(by (6))}\\
&\quad= \frac{\Pr(x)}{\Pr(x) + e^{-\varepsilon_x}(1 - \Pr(x))}\\
&\quad= \frac{\mathbf{c}(x)}{\mathbf{c}(x) + e^{-\varepsilon_x}(n - \mathbf{c}(x))} \quad \text{(by } \mathbf{c}(x) = n\Pr(x)\text{).} \qquad (15)
\end{aligned}$$

By (14) and (15), we have

$$\mathbf{p}_{U|Y=y}(u_i) \leq \frac{e^{\varepsilon_x}}{e^{\varepsilon_x}\mathbf{c}(x) + (n - \mathbf{c}(x))}. \qquad (16)$$

Assume that $\mathbf{c}(x) < (n/\gamma)$. In this case, $e^{\varepsilon_x} = ([\gamma(n - \mathbf{c}(x))]/[n - \gamma\mathbf{c}(x)])$ by (7). Thus, (16) can be written as

$$\begin{aligned}
\mathbf{p}_{U|Y=y}(u_i) &\leq \frac{\gamma(n - \mathbf{c}(x))}{\gamma(n - \mathbf{c}(x))\mathbf{c}(x) + (n - \mathbf{c}(x))(n - \gamma\mathbf{c}(x))}\\
&= \frac{\gamma}{\gamma\mathbf{c}(x) + (n - \gamma\mathbf{c}(x))}\\
&= \frac{\gamma}{n}.
\end{aligned}$$

Since this inequality holds for any $u_i \in \mathcal{U}$, (8) holds.

Assume that $\mathbf{c}(x) \geq (n/\gamma)$. In this case, $\varepsilon_x = \infty$. Thus, (16) can be written as

$$\mathbf{p}_{U|Y=y}(u_i) \leq \frac{1}{\mathbf{c}(x)} \leq \frac{\gamma}{n}.$$

Thus, (8) holds for any $y \in \mathcal{Y}$.

Appendix B
Proof of Proposition 2

Let $x, x' \in \mathcal{X}$ and $y^{(1)}, y^{(2)} \in \mathcal{Y}$. Let $\mathbf{Q}$ be the sequential composition of $\mathbf{Q}^{(1)}$ and $\mathbf{Q}^{(2)}$; i.e.,

$$\mathbf{Q}\left(\left(y^{(1)}, y^{(2)}\right)|x\right) = \mathbf{Q}^{(1)}\left(y^{(1)}|x\right)\mathbf{Q}^{(2)}\left(y^{(2)}|\left(y^{(1)}, x\right)\right).$$

Since $\mathbf{Q}^{(1)}$ provides $\mathcal{E}^{(1)}$-OneID-LDP and $\mathbf{Q}^{(2)}(y^{(1)})$ provides $\mathcal{E}^{(2)}$-OneID-LDP, we have

$$\begin{aligned}
&\mathbf{Q}\left(\left(y^{(1)}, y^{(2)}\right)|x\right)\\
&\quad= \mathbf{Q}^{(1)}\left(y^{(1)}|x\right)\mathbf{Q}^{(2)}\left(y^{(2)}|\left(y^{(1)}, x\right)\right)\\
&\quad\leq e^{\varepsilon_x^{(1)}}\mathbf{Q}^{(1)}\left(y^{(1)}|x'\right)\mathbf{Q}^{(2)}\left(y^{(2)}|\left(y^{(1)}, x\right)\right)\\
&\quad\leq e^{\varepsilon_x^{(1)}}\mathbf{Q}^{(1)}\left(y^{(1)}|x'\right)e^{\varepsilon_x^{(2)}}\mathbf{Q}^{(2)}\left(y^{(2)}|\left(y^{(1)}, x'\right)\right)\\
&\quad= e^{\varepsilon_x^{(1)}+\varepsilon_x^{(2)}}\mathbf{Q}\left(\left(y^{(1)}, y^{(2)}\right)|x'\right)
\end{aligned}$$

which proves Proposition 2.

Appendix C
Proof of Proposition 3

Let $x, x' \in \mathcal{X}$, $y \in \mathcal{Y}$, and $z \in \text{Range}(\lambda)$. Since $\mathbf{Q}$ provides $\mathcal{E}$-OneID-LDP, we have

$$\begin{aligned}
(\lambda \circ \mathbf{Q})(z|x) &= \sum_{y \in \mathcal{Y}} \mathbf{Q}(y|x)\lambda(z|y)\\
&\leq \sum_{y \in \mathcal{Y}} e^{\varepsilon_x}\mathbf{Q}\left(y|x'\right)\lambda(z|y)\\
&= e^{\varepsilon_x}(\lambda \circ \mathbf{Q})\left(z|x'\right)
\end{aligned}$$

which proves Proposition 3.

Appendix D
Proof of Proposition 4

We prove Proposition 4 via the following lemma.

Lemma 1: Let $\varepsilon_x \in \mathbb{R}_{\geq 0}$ be a privacy budget for personal data $x \in \mathcal{X}$. Let $\mathcal{E} = \{\varepsilon_x\}_{x \in \mathcal{X}}$. If a randomized mechanism $\mathbf{Q}$ provides $\mathcal{E}$-MinID-LDP, then $\mathbf{Q}$ also provides $\mathcal{E}$-OneID-LDP.

Proof of Lemma 1: If a randomized mechanism $\mathbf{Q}$ provides $\mathcal{E}$-MinID-LDP, $\mathbf{Q}$ provides $(\mathcal{E}, r)$-ID-LDP, where $r(\varepsilon_x, \varepsilon_{x'}) = \min\{\varepsilon_x, \varepsilon_{x'}\}$ (see Definition 3). This means that for any $x, x' \in \mathcal{X}$ and any $y \in \mathcal{Y}$, we have

$$\mathbf{Q}(y|x) \leq e^{\min\{\varepsilon_x, \varepsilon_{x'}\}}\mathbf{Q}\left(y|x'\right) \leq e^{\varepsilon_x}\mathbf{Q}\left(y|x'\right).$$

Thus, $\mathbf{Q}$ also provides $(\mathcal{E}, r)$-ID-LDP, where $r(\varepsilon_x, \varepsilon_{x'}) = \varepsilon_x$, which means that $\mathbf{Q}$ provides $\mathcal{E}$-OneID-LDP (see Definition 5).

Proposition 4 is immediately derived from Theorem 1 and Lemma 1. Specifically, a randomized mechanism $\mathbf{Q}$ providing

$\mathcal{E}$-MinID-LDP for $\mathcal{E}$ in (7) also provides $\mathcal{E}$-OneID-LDP for $\mathcal{E}$ in (7) (by Lemma 1). Therefore, $\mathbf{Q}$ provides

$$\mathsf{Acc}_{U|Y=y} \leq \frac{\gamma}{n}$$

for any $y \in \mathcal{Y}$ (by Theorem 1).

References

[1] B. Hull et al., "CarTel: A distributed mobile sensor computing system," in *Proc. 4th Int. Conf. Embedded Netw. Sensor Syst. (SenSys)*, 2006, pp. 125–138.

[2] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from GPS trajectories," in *Proc. 18th Int. Conf. World Wide Web (WWW)*, 2009, pp. 791–800.

[3] B. Kaluza, V. Mirchevska, E. Dovgan, M. Lustrek, and M. Gams, "An agent-based approach to care in independent living," in *Proc. 1st Int. Joint Conf. Ambient Intell. (AmI)*, 2010, pp. 177–186.

[4] H. Hino, H. Shen, N. Murata, S. Wakao, and Y. Hayashi, "A versatile clustering method for electricity consumption pattern analysis in households," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 1048–1057, Jun. 2013.

[5] C. Chelmis, J. Kolte, and V. K. Prasanna, "Big data analytics for demand response: Clustering over space and time," in *Proc. IEEE Int. Conf. Big Data (BigData)*, 2015, pp. 2223–2232.

[6] C. Dwork, "Differential privacy," in *Proc. 33rd Int. Conf. Automata Lang. Program. (ICALP)*, 2006, pp. 1–12.

[7] C. Dwork and A. Roth, *The Algorithmic Foundations of Differential Privacy*. Hanover, MA, USA: Now, 2014.

[8] C. Morris. "Hackers Had a Banner Year in 2019." 2020. [Online]. Available: https://fortune.com/2020/01/28/2019-data-breach-increases-hackers/

[9] U. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security (CCS)*, 2014, pp. 1054–1067.

[10] A. G. Thakurta et al., "Learning new words," U.S. Patent 9 594 741, Mar. 14, 2017.

[11] B. Ding, J. Kulkarni, and S. Yekhanin, "Collecting telemetry data privately," in *Proc. 31st Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 3574–3583.

[12] X. Gu, M. Li, L. Xiong, and Y. Cao, "Providing input-discriminative protection for local differential privacy," in *Proc. 36th IEEE Int. Conf. Data Eng. (ICDE)*, 2020, pp. 505–516.

[13] T. Murakami and Y. Kawamoto, "Utility-optimized local differential privacy mechanisms for distribution estimation," in *Proc. 28th USENIX Security Symp. (USENIX Security)*, 2019, pp. 1877–1894.

[14] J. Acharya, K. Bonawitz, P. Kairouz, D. Ramage, and Z. Sun, "Context-aware local differential privacy," in *Proc. 37th Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 52–62.

[15] N. Li, M. Lyu, and D. Su, *Differential Privacy: From Theory to Practice*. London, U.K.: Morgan & Claypool, 2016.

[16] A. Narayanan and V. Shmatikov, "Myths and fallacies of 'personally identifiable information,'" *Commun. ACM*, vol. 53, no. 6, pp. 24–26, 2010.

[17] O. Hasan, B. Habegger, L. Brunie, N. Bennani, and E. Damiani, "A discussion of privacy challenges in user profiling with big data techniques: The EEXCESS use case," in *Proc. IEEE Int. Congr. Big Data (BigData Congr.)*, 2009, pp. 25–30.

[18] "309 Million Facebook User Profiles Sold on Dark Web and Hacker Forums." 2020. [Online]. Available: https://www.securitymagazine.com/articles/92203-million-facebook-user-profiles-sold-on-dark-web-and-hacker-forums

[19] "The EU General Data Protection Regulation (GDPR)." 2016. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj

[20] G. Maldoff. "Top 10 Operational Impacts of the GDPR: Part 8—Pseudonymization." 2016. [Online]. Available: https://iapp.org/news/a/top-10-operational-impacts-of-the-gdpr-part-8-pseudonymization/

[21] P. Zhao, G. Zhang, S. Wan, G. Liu, and T. Umer, "A survey of local differential privacy for securing Internet of Vehicles," *J. Supercomput.*, vol. 76, pp. 8391–8412, Nov. 2020.

[22] T. Murakami and K. Takahashi, "Toward evaluating re-identification risks in the local privacy model," *Trans. Data Privacy*, vol. 14, no. 3, pp. 79–116, 2021.

[23] T. Wang, J. Blocki, N. Li, and S. Jha, "Locally differentially private protocols for frequency estimation," in *Proc. 26th USENIX Security Symp. (USENIX)*, 2017, pp. 729–745.

[24] V. Torra, *Data Privacy: Foundations, New Developments and the Big Data Challenge*. Cham, Switzerland: Springer, 2017.

[25] B. Jayaraman and D. Evans, "Evaluating differentially private machine learning in practice," in *Proc. 28th USENIX Security Symp. (USENIX Security)*, 2019, pp. 1895–1912.

[26] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Proc. IEEE Symp. Security Privacy (S P)*, 2017, pp. 3–18.

[27] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "*L*-diversity: Privacy beyond *k*-anonymity," in *Proc. 22nd Int. Conf. Data Eng. (ICDE)*, 2006, pp. 24–35.

[28] J. Ullman. "Statistical Inference Is Not a Privacy Violation." 2021. [Online]. Available: https://differentialprivacy.org/inference-is-not-a-privacy-violation/

[29] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, and S. Raskhodnikova, "What can we learn privately?" in *Proc. 49th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, 2008, pp. 531–540.

[30] P. C. M. Arachchige, P. Bertok, I. Khalil, D. Liu, S. Camtepe, and M. Atiquzzaman, "Local differential privacy for deep learning," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5827–5842, Jul. 2020.

[31] R. Bassily and A. D. Smith, "Local, private, efficient protocols for succinct histograms," in *Proc. 47th Annu. ACM Symp. Theory Comput. (STOC)*, 2015, pp. 127–135.

[32] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *Proc. 54th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, 2013, pp. 429–438.

[33] G. Fanti, V. Pihur, and U. Erlingsson, "Building a RAPPOR with the unknown: Privacy-preserving learning of associations and data dictionaries," *Proc. Privacy Enhanc. Technol. (PoPETs)*, vol. 2016, no. 3, pp. 1–21, 2016.

[34] J. Imola, T. Murakami, and K. Chaudhuri, "Locally differentially private analysis of graph statistics," in *Proc. 30th USENIX Security Symp. (USENIX Security)*, 2021, pp. 983–1000.

[35] T. Murakami, H. Hino, and J. Sakuma, "Toward distribution estimation under local differential privacy with small samples," *Proc. Privacy Enhanc. Technol. (PoPETs)*, vol. 2018, no. 3, pp. 84–104, 2018.

[36] P. C. M. Arachchige, P. Bertok, I. Khalil, D. Liu, S. Camtepe, and M. Atiquzzaman, "PAAL: A framework based on authentication, aggregation, and local differential privacy for Internet of Multimedia Things," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2501–2508, Apr. 2020.

[37] Y. Zhao et al., "Local differential privacy-based federated learning for Internet of Things," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 8836–8853, Jun. 2021.

[38] Y. Sei and A. Ohusuga, "Differential private data collection and analysis based on randomized multiple dummies for untrusted mobile crowdsensing," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 926–939, Apr. 2017.

[39] K. Peng, M. Li, H. Huang, C. Wang, S. Wan, and K.-K. R. Choo, "Security challenges and opportunities for smart contracts in Internet of Things: A survey," *IEEE Internet Things J.*, vol. 8, no. 15, pp. 12004–12020, Aug. 2021.

[40] M. U. Hassan, M. H. Rehmani, and J. Chen, "Differential privacy in blockchain technology: A futuristic approach," *J. Parallel Distrib. Comput.*, vol. 145, pp. 50–74, Nov. 2020.

[41] D. Desfontaines and B. Pejó, "SoK: Differential privacies," *Proc. Privacy Enhancing Technol. (PoPETs)*, vol. 2020, no. 2, pp. 288–313, 2020.

[42] S. Doudalis, I. Kotsoginannis, S. Haney, A. Machanavajjhala, and S. Mehrotra. "One-Sided Differential Privacy." 2017. [Online]. Available: http://arxiv.org/abs/1712.05888

[43] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," in *Proc. 20th ACM Conf. Comput. Commun. Security (CCS)*, 2013, pp. 901–914.

[44] K. Chatzikokolakis, M. E. André, N. E. Bordenabe, and C. Palamidessi, "Broadening the scope of differential privacy using metrics," in *Proc. 13th Privacy Enhancing Technol. (PETS)*, 2013, pp. 82–102.

[45] A. Cohen and K. Nissim, "Towards formalizing the GDPR's notion of singling out," *Proc. Nat. Acad. Sci. USA*, vol. 117, no. 15, pp. 8344–8352, 2020.

[46] K. Nissim, "Privacy: From database reconstruction to legal theorems," in *Proc. 40th ACM SIGMOD-SIGACT-SIGAI Symp. Principles Database Syst. (PODS)*, 2021, pp. 33–41.

[47] J. Gehrke, M. Hay, E. Lui, and R. Pass, "Crowd-blending privacy," in *Proc. 32nd Annu. Cryptol. Conf. Adv. Cryptol. (CRYPTO)*, 2012, pp. 479–496.

[48] V. Bindschaedler, R. Shokri, and C. A. Gunter, “Plausible deniability for privacy-preserving data synthesis,” *Proc. VLDB Endow.*, vol. 10, no. 5, pp. 481–492, 2017.

[49] U. Erlingsson, V. Feldman, I. Mironov, A. Raghunathan, and K. Talwar, “Amplification by shuffling: From local to central differential privacy via anonymity,” in *Proc. 30th Annu. ACM/SIAM Symp. Discr. Algorithms (SODA)*, 2019, pp. 2468–2479.

[50] B. Balle, J. Bell, A. Gascon, and K. Nissim, “The privacy blanket of the shuffle model,” in *Proc. 39th Annu. Cryptol. Conf. Adv. Cryptol. (CRYPTO)*, 2019, pp. 638–667.

[51] V. Feldman, A. McMillan, and K. Talwar. “Hiding Among the Clones: A Simple and Nearly Optimal Analysis of Privacy Amplification by Shuffling.” 2021. [Online]. Available: https://arxiv.org/abs/2012.12803

[52] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Springer, 2009.

[53] L. D. Brown, T. T. Cai, and A. DasGupta, “Interval estimation for a binomial proportion,” *Stat. Sci.*, vol. 16, no. 2, pp. 101–133, 2001.

[54] R. G. Newcombe, “Two-sided confidence intervals for the single proportion: Comparison of seven methods,” *Stat. Med.*, vol. 17, no. 8, pp. 857–872, 1998.

[55] S. Wallis, “Binomial confidence intervals and contingency tests: Mathematical fundamentals and the evaluation of alternative methods,” *J. Quant. Linguist.*, vol. 20, no. 3, pp. 178–208, 2013.

[56] J. Hartung, G. Knapp, and B. K. Sinha, *Statistical Meta-Analysis With Applications*. Hoboken, NJ, USA: Wiley, 2008.

[57] T. M. T. Do and D. Gatica-Perez, “The places of our lives: Visiting patterns and automatic Labeling from longitudinal smartphone data,” *IEEE Trans. Mobile Comput.*, vol. 13, no. 3, pp. 638–648, Mar. 2014.

[58] D. Yang, B. Qu, J. Yang, and P. Cudre-Mauroux, “Revisiting user mobility and social relationships in LBSNs: A hypergraph embedding approach,” in *Proc. World Wide Web Conf. (WWW)*, 2019, pp. 2147–2157.

[59] F. J. Ordonez, P. de Toledo, and A. Sanchis, “Activity recognition using hybrid generative/discriminative models on home environments using binary sensors,” *Sensors*, vol. 13, no. 5, pp. 5460–5477, 2013.

[60] R. L. S. Torres, D. C. Ranasinghe, Q. Shi, and A. P. Sample, “Sensor enabled wearable RFID technology for mitigating the risk of falls near beds,” in *Proc. IEEE Int. Conf. RFID (RFID)*, 2013, pp. 191–198.

[61] J. Domingo-Ferrer, S. Ricci, and J. Soria-Comas, “Disclosure risk assessment via record linkage by a maximum-knowledge attacker,” in *Proc. 13th Annu. Conf. Privacy Security Trust (PST)*, 2015, pp. 3469–3478.

[62] N. Ruiz, K. Muralidhar, and J. Domingo-Ferrer, “On the privacy guarantees of synthetic data: A reassessment from the maximum-knowledge attacker perspective,” in *Proc. Int. Conf. Privacy Statistical Databases (PSD)*, 2018, pp. 59–74.

[63] H. Shin, S. Kim, J. Shin, and X. Xiao, “Privacy enhanced matrix factorization for recommendation with local differential privacy,” *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 9, pp. 1770–1782, Sep. 2018.

Takao Murakami (Member, IEEE) received the Ph.D. degree from the University of Tokyo, Tokyo, Japan, in 2014.

He joined Hitachi, Ltd., Yokohama, Japan, in 2006, and the National Institute of Advanced Industrial Science and Technology, Tokyo, in 2014, where he is currently a Senior Research Scientist with the Cyber Physical Security Research Center. He is also a Visiting Associate Professor with Rikkyo University, Tokyo. His research interests include differential privacy, privacy in machine learning, and location privacy.

Dr. Murakami received the IEEE TrustCom 2015 Best Paper Award in 2015. He also received the Dean’s Award of Graduate School of Information Science and Technology from the University of Tokyo, for his Ph.D. thesis in 2014. He is a member of IEICE.

Yuichi Sei received the Ph.D. degree in information science and technology from the University of Tokyo, Tokyo, Japan, in 2009.

From 2009 to 2012, he was with Mitsubishi Research Institute, Chiyoda, Japan. He joined the University of Electro-Communications, Tokyo, in 2013, where he is currently a Professor with the Graduate School of Informatics and Engineering. He is also a Fellow Researcher with Mitsubishi Research Institute and an Adjunct Researcher with Waseda University, Tokyo. His current research interests include pervasive computing, privacy-preserving data mining, and software engineering.