This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

EdgeSanitizer: Locally Differentially Private Deep Inference at the Edge for Mobile Data Analytics

Chugui Xu, *Student Member, IEEE*, Ju Ren, *Member, IEEE*, Liang She, *Member, IEEE*, Yaoxue Zhang, *Senior Member, IEEE*, Zhan Qin, *Member, IEEE*, Kui Ren, *Fellow, IEEE*.

***Abstract*—Deep neural networks have been widely applied in various machine learning applications for mobile data analytics in cloud. However, this approach introduces significant data challenges, because the cloud operator can perform deep inferences on the available data. Recent advances in edge computing have paved the way to more efficient and private data processing at the edge of the network for simple tasks and lightweight models, but challenges still remain in building efficient complex models (e.g., deep learning) for edge computing. To tackle these issues, we propose EdgeSanitizer, a deep inference framework based edge computing with local differential privacy for mobile data analytics. EdgeSanitizer leverages deep learning model to conduct data minimization and obfuscates the learnt features by adaptively injecting noise, thereby forming a new protection layer against sensitive inference. We evaluate its performance in terms of data privacy and utility through theoretical analysis and experimental evaluation. The theoretical analysis proves that EdgeSanitizer can provide provable privacy guarantees with a large improvement in utility. And the experimental results demonstrate the robustness of our approach against sensitive inference, as well as its applicability on resource-constrained edge devices.**

***Index Terms*—Deep Inference, Edge Computing, Local Differential Privacy, Mobile Data Analytics.**

I. Introduction

MOBILE devices such as smartphones and wearables are increasingly gaining popularity as platforms for collecting various data. The collected data is transfered to the cloud to benefit from cloud-based data analytics services such as recommendation systems, health monitoring and urban planning. However, complete data offloading to the cloud introduces unforeseeable delay and heavy communication burden [1], [2]. Therefore, sending personal data to the cloud to perform deep inference seems no longer to be an acceptable solution. A better alternative solution should take advantage of the resource capabilities of personal devices and nearby infrastructures to process data locally, which also promotes the emergence of edge computing [3]–[5]. Edge computing leverages the nearby devices/infrastructures to migrate delay-sensitive and context-aware data analytics from the cloud to the edge for mobile applications [6]–[9].

Chugui Xu, Ju Ren and Yaoxue Zhang are with the School of Computer Science and Engineering, Central South University, Changsha, China, 410083. E-mails: {chuguixu, renju, zyx}@csu.edu.cn.

Liang She is with Mobile E-business Collaborative Innovation Center of Hunan Province, Hunan University of Commerce, Changsha, China, 410205. E-mail: sheliang@csu.edu.cn.

Zhan Qin, Kui Ren are with Institute of Cyberspace Research, Zhejiang University, Hangzhou, China, 310058. E-mail: zhan.qin@utsa.edu, kuiren@buffalo.edu.

Corresponding author: *Ju Ren*.

Although data analytics at the edge have enabled context-aware Apps to provide utility for users, the same data can also be used by an adversary to make sensitive inferences, such as speaker identity [10], location tracking and detection of emotional state. Therefore, there exist fundamentally conflicting requirements between protecting privacy of the sensitive information contained in mobile data and guaranteeing utility of the same data for useful inferences. Additionally, excessive data collection from the users can lead to consequences which are unknown to the users, posing great challenges to preserving data privacy at the edge [11], [12]. Traditional solutions relying on performing complete analytics by local processing, or encryption-based methods are infeasible because of resource limitations and heavy overhead [13], [14]. Therefore, how to conduct privacy-preserving and useful inferences at the edge becomes appealing.

Recently, one predominant technique to address the problem for satisfying differential privacy (DP) [15], [16] with the local setting, called local differential privacy (LDP) [17], has been presented. LDP works by injecting random noise into the released data to achieve DP and ensure that the adversary cannot infer any particular record without relying on a trusted third party, even if the adversary possesses all the remaining records. For example, engineers from Google developed RAPPOR [18], which enables Chrom to collect users' data such as the default search engine of the browser, to capture malicious hijacking of user settings. However, LDP may increase the magnitude of added noise while decreasing the data utility. Since useful inferences mean that we can extract desired knowledge from the collected data, we have no need to transfer all of personal data to edge devices. An alternative way of protecting mobile data is to build a lightweight, LDP-based data minimization model to minimize the amount of data while maximizing the data utility.

Applying LDP at the edge can enable applications to collect some inaccessible data with strict privacy guarantees. The increased amount of data will significantly improve the performance of some learning tasks. Therefore, to explore the inherent structural characteristics of mobile data, it is significant to combine deep learning and local differential privacy to extract a compact feature representation for useful inferences. Shokri and Shmatikov propose a differentially private deep neural network (DP-ML) by adding random noise into gradients to guarantee privacy [19]. However, the disadvantage of DP-ML is that all parameters are treated equally [19]–[21]. This approach may be infeasible in real applications, because different features and parameters nor-

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

mally have different impacts on the model output. Therefore, It is urgent to design a lightweight useful inference mechanism with the minimum features, to make it apply in various deep neural networks while maximizing the utility.

To this end, we present EdgeSanitizer, a deep inference framework based on edge computing with local differential privacy for mobile data analytics. EdgeSanitizer leverages deep learning model to conduct data minimization to limit the size of data. Also, EdgeSanitizer can obfuscate the learnt features extracted from the raw data by adaptively injecting noise to achieve LDP, thereby forming a new protection layer against sensitive inference. Summarily, our contributions of this paper are as follows.

(i) We propose a deep inference framework based on edge computing with local differential privacy for mobile data analytics to tune the trade-off between privacy and utility of data.

(ii) We develop a new lightweight technique to automatically extract features relevant to useful inferences by extending deep learning models for data minimization, and obfuscate the learnt features by adaptively injecting noise to achieve LDP, thereby forming a new protection layer against sensitive inference at the edge.

(iii) We theoretically prove that EdgeSanitizer can satisfy ε-LDP while guaranteeing the utility of mobile data. Moreover, we can further derive the error upper bounds of EdgeSanitizer.

(iv) We comprehensively evaluate the performance of EdgeSanitizer by data privacy and utility. We also demonstrate the robustness of our approach against sensitive inference, as well as its feasibility by conducting performance evaluation on a representative resource-constrained edge device (i.e., a smartphone).

The remaining of the paper is organized as follows. We introduce the preliminaries, and present the framework of EdgeSanitizer and problem statement in section III. Section IV introduces the design of EdgeSanitizer and section V performs the theoretical analysis on the privacy and utility guarantees of EdgeSanitizer. Section VI shows experimental results. Then we introduce the related works in section VII, followed by a conclusion in section VIII.

II. Preliminaries

We briefly review the concept and some important properties of Local differential privacy (LDP) and deep learning in this section. The symbols frequently used in this paper are listed in Table I.

TABLE I: Frequently Used Symbols

Symbol	Description
ϵ	Privacy budget
$X, \hat{X}$	Input database and output database
D_1, D_2	Any two neighboring datasets
$L(\cdot), \hat{L(\cdot)}$	The loss functions
$x_t, x_t^{'}$	Vectors in datasets
R	Mobile data matrix
β_i	Privacy budget ratio
N, T	The number of mobile devices and timestamps
$f_t, f_t^{'}$	Features in datasets
$Q(\cdot)$	The query function
$h, h^{'}$	Hidden neurons
ω	Attribute value
$EN_{nc}(\cdot)$	Non-linear encoding function
E_{re}	The reconstruction error
F	The minimax filter
$dim(\cdot), d$	Dimension function and the number of feature dimension
$L^{*U/S}$	The cost function for useful and sensitive inferences
LPM	Laplace Perturbation Mechanism
b	The static bias
θ	a fixed but possibly unknown state of training

A. Local Differential Privacy

Differential privacy (DP) [15] is a rigorous privacy framework that prevents an attacker from inferring a particular record in a statistical database [16]. DP randomizes the query results, computed over the multi-user database, to ensure that the risk to an individual record's privacy does not increase substantially (bounded by a function of the privacy budget ε) as a result of participating in the database. Local differential privacy (LDP) [17] is defined under the setting where the user does not trust anyone (not even the central data collector). LDP adopts randomized response method to provide plausible deniability for individuals responding to sensitive inference. Specifically, ε-LDP is defined as follows:

Definition 1. *(ε-LDP) [17] A randomized algorithm $\mathcal{A}(\cdot)$ provides ε-LDP if for any two databases D_1, D_2 and for any output set $\mathcal{O}$,*

$$\max_{D_1, D_2, O} \frac{P(\mathcal{A}(D_1) = \mathcal{O})}{P(\mathcal{A}(D_2) = \mathcal{O})} \leq \exp(\varepsilon) \tag{1}$$

where $\mathcal{A}(D_1)$ (resp.$\mathcal{A}(D_2)$) is the output of $\mathcal{A}(\cdot)$ on input D_1 (resp.D_2) and ε is the privacy budget. Smaller value of ε corresponds to a higher privacy level.

To enable perturbation mechanisms in LDP for satisfying differential privacy, we adopt sequential composability of differential privacy elaborated by McSherry in [22].

Theorem 1. *Given t random mechanisms $\mathcal{A}_i\,(1 \leq i \leq t)$, each of which satisfies ε_i-differential privacy. Then, the sequence of $\mathcal{A}_i\,(D)$ satisfies $\sum_{i=1}^{t} \varepsilon_i$-differential privacy,*

where $\mathcal{A}_i$ can be arbitrary functions of the input database and preceding outputs.

B. Deep Learning

Deep Learning is to learn multiple levels of representation and abstraction that help to make use of data such as text, image and video [19]. Deep neural networks(DNN) are effective architectures for most deep learning tasks such as speech recognition and computer vision. Since the large number of layers in the DNN, each training is transmitted layer by layer from back to front. If the transfer term is less than 1, the gradient may become very small and tend to 0, namely vanishing gradients problem, so that the network can be trained with little change. To tackle this problem, The softmax activation function and log likelihood loss function are usually used in the output layer.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

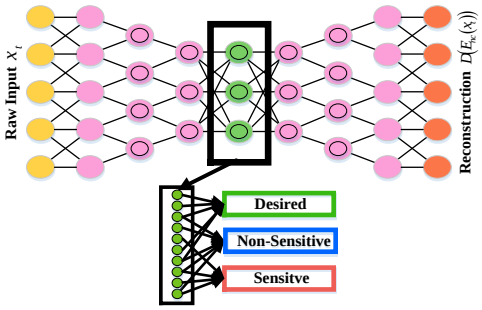


Fig. 1: Inference Based on Deep Neural Network

In this paper, we define a loss function $L(\cdot)$ that denotes the penalty for mismatching the training data as follows,

$$L(\theta) = \frac{1}{N}\sum_{i} L(\theta, x_i), \tag{2}$$

where the loss $L(\theta)$ denotes the average of the loss over the training samples $x_1, ..., x_N$. The goal of training is to find θ and yield an acceptably small loss.

In this paper, we investigate two kinds of inference based on deep neural network as shown in Fig. 1: sensitive inference and useful inference. In deep inference, a fraction of data can be used to infer sensitive data and the remaining desired data can be used for useful inference [23].

III. Framework and Problem Statement

A. The Architecture of EdgeSanitizer

Suppose that we plan to utilize an edge server to infer some useful data of interest (e.g., body count, trajectory). In the same way, we should prevent the exposure of sensitive data (e.g., identity) to the adversary. Hence, the data shared with the edge server should possess two important properties: (i) inferring the useful data is possible; and (ii) deducing the sensitive data is not possible. To achieve these goals, we obfuscate the data by adaptively injecting random noise. Then, we can transfer this noisy data to the cloud for further analytics, without initial privacy concerns. Therefore, we propose EdgeSanitizer, a deep inference framework based on edge computing with LDP for mobile data analytics.

The framework of EdgeSanitizer is shown in Fig. 2. It includes three key technologies: i) *data minimization* based on deep learning to extract features representation for useful inference, ii) *data obfuscation* with LDP to obfuscate data by adaptively injecting random noise, and iii) *data reconstruction* to reassemble the perturbed features for data analytics. The three key technologies enable EdgeSanitizer to preserve the privacy while maximizing the utility of mobile data. To extract compact features representation of the data for useful inference, we design an autoencoder by exploring the inherent structural properties of the mobile data. However, there are still some extraction features associated with sensitive inference. Therefore, we adopt LDP to obfuscate the extracted features by injecting random noise. Finally, we reconstruct the obfuscated features for mobile data analytics. The implementation details of EdgeSanitizer is described in section IV.

B. Problem Statement

For a mobile sensing system, consisting of N mobile devices across T timestamps, the temporal mobile data matrix $R \in \mathbb{R}^{N_S \times T}$ is a real matrix where $R(i,j)$ records the sensing data of the i-th sensor at the j-th timestamp [24]. The mobile data is generated by measurements of physical value such as temperature and humidity for various mobile application. To fully explore the temporal characteristics of the data, we follow common practice [25] and partition it into a series of segments according to a user-specified time window size N_w. For the w-th window, we stack the corresponding columns within it to form a column vector which is denoted by $x_t \in \mathbb{R}^{N_s N_w \times 1}$. The temporal mobile sensor matrix can thus be reformulated as $X = [x_1, x_2, ...]$, where $X \in \mathbb{R}^{N_s N_w \times \frac{T}{N_w}}$. For example, collecting different physiological data using heart rate sensor and blood pressure sensor can generate time-series for monitoring users' health status [26]. However, many malicious Apps can secretly collect personal data to infer sensitive data about individual's activities [27].

To address this problem, our objective is to design tailored techniques to achieve that i) an edge server is able to collect the personal data from each mobile while satisfying LDP; and ii) from the collected data, the edge server can extract features relevant to useful inferences by extending the deep learning models for data minimization.

IV. The Design of EdgeSanitizer

In this section, we introduce the design details of EdgeSanitizer. EdgeSanitizer consists of three key steps: i) extracting features relevant to the useful inferences by extending the deep learning models for data minimization, and ii) obfuscating these features by adaptively injecting noise to achieve LDP guarantees, and then iii) reconstructing data from all the mobile devices by private distribution estimation. In the following, we discuss the procedures of each step in detail.

A. Data Minimization based on Deep Learning

Mobile data is composed of many high-dimensional attributes. These features are extracted by deep learning to minimize the amount of data. To decrease communication cost, the edge server should reduce the dimensionality of mobile data by applying autoencoder. Additionally, data minimization [28] is to protect privacy by limiting the collection of data to the minimum extent.

The features extraction of mobile data consists of four phases:

- Feature Transformation: Assume that each feature in dataset X has a domain Ω. Each candidate feature value $\omega \in \Omega$ is transformed into a k-bit string S by several hash functions. Only if the number of hash functions and k are selected, the transformation can leverage a bit string S to represent any feature $\omega \in \Omega$.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

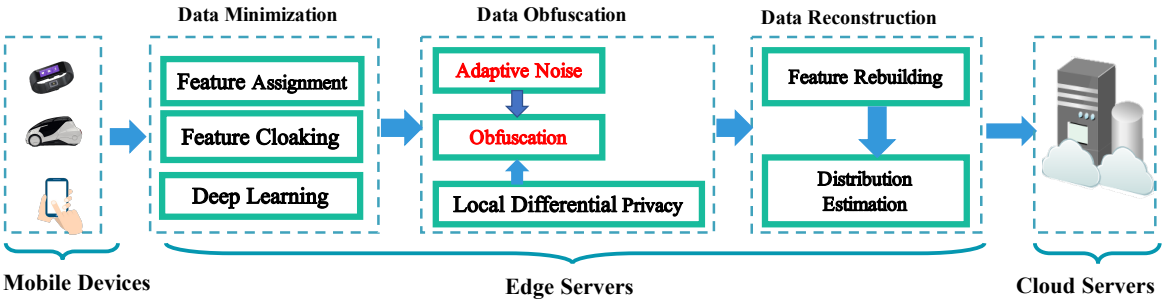


Fig. 2: The Framework of EdgeSanitizer

- Feature Randomization: In this phase, each bit will be randomized by the minimax filter F to 0 or 1 to prevent sensitive inference.
- Constructing Data Minimization Model: The data representation is to learn multi-layer transformations from the raw data to feature representations by an autoencoder. The autoencoder is to capture potential factors and discover a robust and useful feature set from raw data and encodes the input x_t into feature representation $y = EN_{nc}(x_t)$, so that the output $z = D(EN_{nc}(x_t))$ is a reconstruction of extracted feature. The autoencoder aims to minimize the error E_t with respect to x_t, n and t.

$$\min E_t = \min \sum_{t=1}^{T/n} E_{re}\left(x_t, D\left(EN_{nc}\left(x_t\right)\right)\right), \quad (3)$$

  where $E_{re}\left(x_t, D\left(EN_{nc}\left(x_t\right)\right)\right)$ is the reconstruction error between the input data x_t and its reconstructed output $D\left(EN_{nc}\left(x_t\right)\right)$.
- Incorporating Useful Inference: The objective of deep inference is to extract some features from the mobile data for useful inference. Therefore, we modify the deep inference models through incorporating the useful inferences. The mathematical formulation of the autoencoder model [29] is

$$\begin{aligned} \min \theta = \min & \sum_{t=1}^{T/N} L\left(x_t, D_{ec}\left(E_{nc}\left(x_t\right)\right)\right) \\ & + \lambda \sum_{t,i} w_{t,i}^2 + \delta \sum_{i=1}^{d_f} KL\left(\rho \parallel \hat{\rho}_i\right) \end{aligned} \quad (4)$$

  where the encoder function $E_{nc}(\cdot)$ maps the input data $x_t \in R^{d_{x_t} \times 1}$ to the hidden features $f_t \in R^{d_{f_t}}$ according to $f_t = E_{nc}\left(x_t\right)$, and the decoder function $D_{ec}(\cdot)$ maps the outputs of the hidden features back to the original input space according to $\tilde{x}_t = D_{ec}\left(f_t\right)$. $L\left(u, v\right)$ is a loss function, typically the square loss $L\left(u, v\right) = \| u - v\|^2$. $KL\left(\rho \parallel \hat{\rho}_i\right)$ is the KL divergence between two Bernoulli random variables with mean ρ and $\hat{\rho}_i$ respectively.

Then we analyze the cost function for each inference using an android smartphone as the edge server in the case study, where behavior-based authentication is considered as a useful inference and activity mode detection is deemed sensitive. Both the useful and sensitive inferences can be mathematically transformed into a classification problem, which can be addressed by machine learning techniques. For instance, by leveraging the popular ridge regression technique [30], we can learn an optimal classifier as follows.

$$\begin{aligned} C^* &= \arg\min_c L^{*U/S} \\ &= \arg\min_c \beta \|c\| + \sum_{t=1}^{T/n} \left(c^T x_t - y_t^*\right) \end{aligned} \quad (5)$$

where $L^{*U/S}$ represents the cost function for the useful and sensitive inferences respectively. For behavior-based useful inference, we have the label $y_t^* \in \{1, -1\}$, where 1 represents the legitimate user and -1 represents the adversary. The optimal classifier C^* learned can be utilized to label the new mobile sensor data for behavior-based authentication or activity mode detection.

B. Locally Differentially Private Deep Inference with Adaptive Noise

To achieve ε-LDP, we design effective perturbation mechanisms to achieve ε-LDP for protecting mobile data in a single-user setting. Our key insight is to exploit the structural characteristics of mobile sensor data to enhance privacy-utility tradeoffs. The Laplace Perturbation Mechanism (LPM) [15] applies noise drawn from a suitable Laplace distribution to perturb the query results. More formally, for a query function $Q(\cdot)$, LPM computes and outputs $A(D) = Q(D) + Lap(\lambda)$, where $\lambda = \triangle Q/\varepsilon$ is the parameter of the Laplacian noise and $\triangle Q = \max_{D_1, D_2} \parallel Q(D_1) - Q\left(D_2\right) \|_1$ is the global sensitivity of $Q(\cdot)$.

When applying LDP on segmented mobile sensor data x_t, the neighboring databases x_{t1}, x_{t2} may differ in all their possible tuples (i.e., all sensor recordings across all timestamps within the same window), while the neighboring databases in traditional DP frameworks only differ in one tuple. Intuitively, according to the composition theorem of DP [22], the baseline approach is to insert a Laplacian noise to each temporal mobile data point with the same parameter of $\lambda = dim(x_t)\triangle Q/\varepsilon$ to achieve ε-LDP, where $dim(x_t)$ is the dimension of each segmented mobile data x_t. Therefore, The approach is feasible when each extracted feature has an equal contribution to the output.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

Generally, we adopt activation functions such as sigmoid, and the deep inference of a hidden neuron h can be denoted as

$$h_{x_t}(W) = b + x_t W^T \tag{6}$$

where W is the weight of h_{x_t} and b is a static bias.

Actually, the assumption is not valid. Because the relevances are different. The differentially private relevances are also different. Therefore, differential privacy deep inference by injecting the same noise into all extracted features might affect the utility of mobile data.

To address this issue, we obfuscate the extracted features with adaptive noise. For hidden units $h_{x_t}(W)$, we adaptively inject more noise into extracted features which are less relevant to the output. Consequently, we incorporate a privacy budget ε_i and privacy budget ratio β_i for each i-th extracted feature as follows

$$\begin{cases} \beta_i = \frac{d \times |\bar{x}_i|}{\sum_{j=1}^{d} |\bar{x}_i|} \\ \varepsilon_i = \beta_i \times \varepsilon \end{cases} \tag{7}$$

where x_i denotes the average relevances of all the i-th input features.

We define $\triangle h_0 = \sum_{h \in h_0} d$, and β_j is a part of the contribution to $\triangle h_0$ from the j-th extracted feature to the neuron $h \in h_0$. For a random training batch L, each extracted feature f_i of every neuron h in the first layer h_0 is obfuscated by injecting random noise:

$$x_t^{'} = x_i + \frac{1}{|X|} Lap\left(\triangle h_0 / \varepsilon_j\right) \tag{8}$$

where $x_t^{'}$ denotes the perturbed input features. Then we build a differentially private deep inference layer $h_0^{'}$, which includes obfuscated neurons $h_X^{'}(W)$:

$$h_0^{'}(W_0) = \left\{ h_L^{'}(W) \right\}_{h \in h_0} \tag{9}$$

$$h_X^{'}(W) = \sum_{x_t \in X} \left(x_t^{'} W^T + b^{'} \right) \tag{10}$$

where $b^{'} = b + \frac{1}{|X|} Lap\left(\frac{\triangle h_0}{\varepsilon}\right)$ is the obfuscated bias. The Lemma 1 derives that the bound of $\triangle h_0$ is $2\sum_{h \in h_0} d$.

Lemma 1. *Let X and $X^{'}$ be any two neighboring datasets. Supposed that $h_0 X$ and $h_{0X'}$ is the first layer on X and $X^{'}$. Then, we have the inequality as follows:*

$$\triangle h_0 = \sum_{h \in h_0} \sum_{j=1}^{d} \left\| \sum_{x_t \in X} x_{tj} + \sum_{x_t^{'} \in X'} x_{tj}^{'} \right\| \leq 2 \sum_{h \in h_0} d,$$

where d is the number of features.

Proof. Suppose that X and $X^{'}$ differ in a feature, and x_n ($x_n^{'}$) is a feature in $X(X^{'})$. For $\forall x_t, j : x_{tj} \in [0,1]$, we have that:

$$\begin{aligned} \triangle h_0 &= \sum_{h \in h_0} \sum_{j=1}^{d} \left\| \sum_{x_t \in X} x_{tj} + \sum_{x_t^{'} \in X'} x_{tj}^{'} \right\| \\ &= \sum_{h \in h_0} \sum_{j=1}^{d} \left\| x_{tj} + x_{tj}^{'} \right\| \leq 2 \max_{x_t \in X} \sum_{h \in h_0} \sum_{j=1}^{d} \|x_{tj}\| \\ &\leq 2 \sum_{h \in h_0} d, \end{aligned}$$

which completes the proof. □

LDP considers the worst-case adversary which can rigorously protect against all possible inferences computed over the data (recall Definition 1). In the absence of a user-specified set of sensitive inferences, or otherwise if the user chooses to operate under the LDP guarantees, we develop our perturbation mechanism through perturbing the features learnt from the deep learning based data minimization. Formally, to achieve ε-LDP, EdgeSanitizer inserts a Laplacian noise with parameter $\lambda = \frac{1}{|X|} Lap\left(\triangle h_0 / \varepsilon_j\right)$ to each previously learned feature. EdgeSanitizer mechanism is summarized in Algorithm 1, which satisfies rigorous ε-LDP as will be discussed in Theorem 1.

Algorithm 1: Deep Inference with Local Differential Privacy

Input: original mobile data $\{x_t\}_{t=1}^{T/N_w}$; hidden layers H, Privacy Budget ε

Output: Perturbed mobile Data $\left\{ x_t^{'} \right\}_{t=1}^{T/N_w}$

```
for each t = 1, 2, ..., T/N_w do
    Extract features f_t from x_t by data minimization mechanism;
    Inject ramdom noise into weights of the deep inference layer h_0;
    △h_0 ≤ Σ_{h∈h_0} d;
    for j ∈ [1, d] do
        ε_i = β_i × ε;
    for f_t ∈ X, j ∈ [1, d] do
        Obtain perturbed features:
        x'_t = x_t + (1/|X|) Lap(△h_0/ε_j);
    Reconstruct perturbed mobile data X'_t = Φ(x'_t);
Return {X'_t}_{t=1}^{T/N_w}.
```

Our mechanism is different from the baseline approach because we add Laplacian noise after the application of the deep learning based data minimization, while the baseline approach directly adds Laplacian noise to the raw mobile data without the deep learning mechanism. After perturbing these features, we reconstruct the perturbed mobile data according to the decoder function $D_{ec}(\cdot)$ in autoencoder. Note that our privacy objective is also different from that in [19], [20], since they aim to protect each user's training data in the deep

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

learning training stage under the multi-user settings while in contrast we aim to protect the privacy of mobile data stream in single-user settings.

C. Private Distribution Estimation

To achieve LDP distribution estimation, EdgeSanitizer consist of two step: i) extracting features with modified autoencoder and ii) injecting random noise [31]. Consequently, edge server can estimate the distribution from different mobile devices by deep learning.

At the edge server, we adopt modified autoencoder to extract feature from different mobile devices and estimate the distribution. All extracted features will be transformed into bitwise. The privacy can be guaranteed by the edge server with LDP by adaptively injecting random noise. The label count for a dataset is denoted by $j \in [m]$ and an input feature is denoted as $x_t : n_j(x_t) = |\{i : i \in [n], f_i(x_t) = j\}|$. Considering the modified autoencoder as the feature vector, the edge servers can estimate the distribution of single feature by deep inference. If we adopt the largest label, the converged result may rely on the crowdsourced data from different devices.

In this way, if features are mutual-independent, we can infer that the combinations of features from different mobile devices are also mutual-independent. Therefore, when autoencoder of each feature are mutual-independent, the output of autoencoders of different features is mutual-independent as well.

V. Theoretical Analysis

In the section, we perform the theoretical analysis in terms of the privacy and utility guarantees of EdgeSanitizer.

A. Privacy Guarantee

According to the following theorem, we can prove that our perturbation mechanisms summarized in Algorithms 1 satisfy ε-LDP.

Theorem 2. *Algorithm 1 satisfies ε-LDP.*

Proof. Assuming that each $h \in h'_{0X}$, h can be expressed as

$$h'_X(W) = \sum_{j=1}^{d}\left[\sum_{x_t\in X}\left(x_t + \frac{1}{|X|}Lap\left(\frac{\triangle h_0}{\varepsilon_j}\right)\right)W^T\right] + \sum_{x_t\in X}\left(b + \frac{1}{|X|}Lap\left(\frac{\triangle h_0}{\varepsilon}\right)\right)$$

Suppose the static bias $b = 1$ as the 0-th extracted feature and its relevant weight W_b, i.e., $x_{t0} = b = 1$ and $W = W_b \cup W$. Then, we have

$$\begin{aligned} h'_X(W) &= \sum_{j=1}^{d}\left[\sum_{x_t\in X}\left(x_t + \frac{1}{|X|}Lap\left(\frac{\triangle h_0}{\varepsilon_j}\right)\right)W^T\right] \\ &= \sum_{j=1}^{d}\left[\sum_{x_t\in X} x_t + Lap\left(\frac{\triangle h_0}{\varepsilon_j}\right)\right]W^T \\ &= \sum_{j=0}^{d}\phi_j^h W^T, \end{aligned}$$

where $\phi_j^h = \sum_{x_t\in X} x_t + Lap\left(\frac{\triangle h_0}{\varepsilon_j}\right)$.

Consequently, ϕ_j^h is the obfuscation of extracted feature x_t with the j-th weight $W_j \in W$ of the hidden neuron h on X. We have

$$Pr\left(h'_{0X}(W_0)\right) = \prod_{h\in h_0}\prod_{j=0}^{d} exp\left(\frac{\varepsilon_j\left\|\sum_{x_t\in X} x_{tj} - \phi_j^h\right\|}{\triangle h_0}\right).$$

$\triangle h_0$ is set to $2\sum_{h\in h_0} d$, and $h'_X(W_0)$ is the output. Let X and X' be any two neighboring batches. Given parameter W_0, we have

$$\begin{aligned} \frac{Pr\left(h'_{0X}(W_0)\right)}{Pr\left(h'_{0X'}(W_0)\right)} &= \frac{\prod_{h\in h_0}\prod_{j=0}^{d} exp\left(\frac{\varepsilon_j\left\|\sum_{x_t\in X} x_{tj} - \phi_j^h\right\|}{\triangle h_0}\right)}{\prod_{h\in h_0}\prod_{j=0}^{d} exp\left(\frac{\varepsilon_j\left\|\sum_{x'_t\in X'} x_{tj} - \phi_j^h\right\|}{\triangle h_0}\right)} \\ &\le \prod_{h\in h_0}\prod_{j=0}^{d} exp\left(\frac{\varepsilon_j}{\triangle h_0}\left\|\sum_{x_t\in X} x_{tj} - \sum_{x'_t\in X'} x_{tj}\right\|\right) \\ &\le \prod_{h\in h_0}\prod_{j=0}^{d} exp\left(\frac{\varepsilon_j}{\triangle h_0} 2\max_{x_n\in X}\|x_{nj}\|\right) \\ &\le \prod_{h\in h_0}\prod_{j=0}^{d} exp\left(\frac{2\varepsilon_j}{\triangle h_0}\right) \\ &\le \prod_{h\in h_0}\prod_{j=0}^{d} exp\left(\varepsilon\frac{2\frac{d\times R'_j}{\sum_{j=1}^{d}|R_j|}}{\triangle h_0}\right) \\ &\le exp\left(\varepsilon\frac{2\sum_{h\in h_0} d\left[\sum_{j=1}^{d}\frac{\left|R'_j\right|}{\sum_{j=1}^{d}|R_j|}\right]}{\triangle h_0}\right) = exp(\varepsilon). \end{aligned}$$

Therefore, Algorithm 1 satisfies ε-LDP. □

B. Utility Guarantee

For a perturbation algorithm A, let us denote $Error(A) = E_A[\|A(D) - D\|_1]$ as the expected error in the release of data D, where $E_A[\cdot]$ is the expectation taken over the randomness of A. We quantify the utility advantage of EdgeSanitizer over the baseline approach in Theorem 3.

Theorem 3. *For EdgeSanitizer (corresponding to Algorithm 1), the expected error $E(A)$ is lower than that of the baseline approach by a factor of $\frac{dim(x_t)}{dim(f_t)}$, where f_t is the feature set extracted from the segmented mobile data x_t by using our deep inference based data minimization approach.*

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

Proof. First, we derive the variance of a randomized algorithm $A(\cdot)$ for our EdgeSanitizer as

$$Var\left(\left(A\left(x_t\right)\right)\right) = \sum_{i=1}^{dim(f_t)} \frac{Var\left(f_t^{'}\right)}{dim(x_t)^2} \leq \frac{dim(f_t)^2(\triangle f_t(i))^2/\varepsilon^2}{dim(x_t)^2} \leq \frac{dim(f_t)^2(\triangle Q)^2/\varepsilon^2}{dim(x_t)^2} \leq \frac{dim(f_t)^2(\triangle Q)^2}{dim(x_t)^2\varepsilon^2}.$$

Then, we compute the expected error as

$$\begin{aligned} &E\left[\left|A\left(x_t\right) - x_t\right|\right] \\ &\leq E\left[\left\|A\left(x_t\right) - E\left(A\left(x_t\right)\right)\right\|_1\right] + E\left[\left\|A\left(x_t\right) - x_t\right\|_1\right] \\ &= E_{re}\left(A\left(\left(x_t\right)\right)\right) + \sqrt{E\left[\left\|A\left(x_t\right) - x_t\right\|_2^2\right]} \\ &= E_{re}\left(A\left(\left(x_t\right)\right)\right) + \sqrt{Var\left(A\left(x_t\right)\right)}. \end{aligned}$$

Note that our deep inference based data minimization mechanism would result in a negligible reconstruction error, which makes it fairly applicable in many practical scenarios. It is likely that the reconstruction error $E_{re}\left(A\left(x_t\right)\right)$ is much lower than the perturbation error caused by adding noise. Therefore, we approximate the utility performance of EdgeSanitizer as $E\left(A\right) \approx \frac{dim(f_t)\triangle Q}{dim(x_t)\varepsilon}$. Similarly, we evaluate the expected error for the baseline approach as $E\left[\left\|LPM\left(x_t\right) - x_t\right\|_1\right] = E\left[\left\|Lap\left(\triangle Q/\varepsilon\right)\right\|_1\right] = \triangle Q/\varepsilon$. Comparing the utility performance for both methods, we can easily find that EdgeSanitizer can reduce the expected error of the baseline approach with a factor of $\frac{dim(x_t)}{dim(f_t)}$. □

Therefore, according to Lemma 1, $\triangle h_0$ is dependent of d, and $\triangle L$ is only dependent of the number of neurons in the output layer and the last hidden layer. However, $\triangle h_0$ and $\triangle L$ are independent of the number of iteration. Moreover, the average error incurred by EdgeSanitizer is bounded by $T \times \frac{e^2+2e-1}{e(1+e)^2}$ (the proof refers to Theorem 4).

Theorem 4. *Given two polynomial functions $L_X(\theta)$ and $\tilde{L}_X(\theta)$, the average error of the approximation is always bounded as follows*

$$\left|L_X\left(\hat{\theta}\right) - L_X\left(\tilde{\theta}\right)\right| \leq T \times \frac{e^2+2e-1}{e(1+e)^2}$$

where $\hat{\theta} = arg\min_\theta L_X(\theta)$ and $\tilde{\theta} = arg\min_\theta \tilde{L}_X(\theta)$.

Proof. Suppose that

$$\begin{aligned} \hat{\theta} &= \arg\min_\theta L_X(\theta), \\ \tilde{\theta} &= \arg\min_\theta \tilde{L}_X(\theta), \\ \alpha &= \max_\theta\left(L_X(\theta) - \tilde{L}_X(\theta)\right), \\ \beta &= \min_\theta\left(L_X(\theta) - \tilde{L}_X(\theta)\right). \end{aligned}$$

We have $\alpha \geq L_X\left(\hat{\theta}\right) - \tilde{L}_X\left(\hat{\theta}\right)$ and $\beta \leq L_X(\theta^*) - \tilde{L}_X(\theta^*)$. Therefore, we have

$$\begin{aligned} &L_X\left(\hat{\theta}\right) - \tilde{L}_X\left(\hat{\theta}\right) - L_X(\theta^*) + \tilde{L}_X(\theta^*) \leq \alpha - \beta. \\ \leftrightarrow\ &L_X\left(\hat{\theta}\right) - L_X(\theta^*) \leq \alpha - \beta + \left(\tilde{L}_X\left(\hat{\theta}\right) - \tilde{L}_X(\theta^*)\right). \end{aligned}$$

In addition, $\tilde{L}_X\left(\hat{\theta}\right) - \tilde{L}_X(\theta^*) \leq 0$, so $L_X\left(\hat{\theta}\right) - L_X(\theta^*) \leq \alpha - \beta$. If $a \geq 0$ and $\beta \leq 0$, then we have

$$\left|L_X\left(\hat{\theta}\right) - L_X(\theta^*)\right| \leq \alpha - \beta$$

The inequality shows that the error incurred by truncating the Taylor series approximate function depends on the maximum and minimum values of $L_X(\theta) - \tilde{L}_X(\theta)$. To quantify the magnitude of the error, we first rewrite $L_X(\theta) - \tilde{L}_X(\theta)$ as

$$L_X(\theta) - \tilde{L}_X(\theta) = \sum_{x=1}^{T}\left[L_X\left(W_{x(k)}\right) - \tilde{L}_X\left(W_{x(k)}\right)\right]$$

Let $\omega_x \in [\omega_{qx} - 1, \omega_{qx} + 1]$. According to the well-known result in [32], $\frac{1}{|X|}\left(L_X\left(W_{x(k)}\right) - \tilde{L}_X\left(W_{x(k)}\right)\right)$ must be in the interval $\left[\sum_q \frac{\min_{\omega x} f_{qx}^3(\omega_x)(\omega_x-\omega_{qx})^3}{6}, \sum_x \frac{\max_{\omega x} f_{qx}^3(\omega_x)(\omega_x-\omega_{qx})^3}{6}\right]$.

If $\sum_q \frac{\min_{\omega x} f_{qx}^3(\omega_x)(\omega_x-\omega_{qx})^3}{6} \leq 0$, and $\sum_x \frac{\max_{\omega x} f_{qx}^3(\omega_x)(\omega_x-\omega_{qx})^3}{6} \geq 0$, then we have that: $\left|\frac{1}{|X|}\left(L_X(\theta) - \tilde{L}_X(\theta)\right)\right| \leq \sum_{x=1}^{T}\sum_q \frac{\max_{\omega x} f_{qx}^3(\omega_x)(\omega_x-\omega_{qx})^3 - \min_{\omega x} f_{qx}^3(\omega_x)(\omega_x-\omega_{qx})^3}{6}$.

This analysis applies to the case of the cross-entropy error-based loss function as follows. First, for the functions

$$\begin{aligned} f_{1x}^3(\omega_{1x}) &= \frac{2y_{ix}e^{\omega_{1x}}}{\left(1+e^{\omega_{1x}}\right)^3}, \\ f_{2x}^3(\omega_{2x}) &= (1-y_{ix})\frac{e^{-\omega_{2x}}\left(e^{-\omega_{2x}}-1\right)}{\left(1+e^{\omega_{2x}}\right)^3}, \end{aligned}$$

We have

$$\begin{aligned} arg\min_{\omega_{1x}} f_{1x}^3(\omega_{1x}) &= \frac{-2e}{(1+e)^3} < 0, \\ arg\max_{\omega_{1x}} f_{1x}^3(\omega_{1x}) &= \frac{2e}{(1+e)^3} > 0, \\ arg\min_{\omega_{2x}} f_{2x}^3(\omega_{2x}) &= \frac{1-e}{e(1+e)^3} < 0, \\ arg\max_{\omega_{2x}} f_{2x}^3(\omega_{2x}) &= \frac{e(e-1)}{(1+e)^3} > 0. \end{aligned}$$

Thus, the average error of the approximation is at most

$$\begin{aligned} &\left|L_X\left(\hat{\theta}\right) - L_X\left(\tilde{\theta}\right)\right| \leq \\ &T \times \left[\left(\frac{2e}{(1+e)^3} - \frac{-2e}{(1+e)^3}\right) + \left(\frac{e(e-1)}{(1+e)^3} - \frac{1-e}{e(1+e)^3}\right)\right] \\ &= T \times \frac{e^2+2e-1}{e(1+e)^2}, \end{aligned}$$

which completes the proof. □

VI. Experiment and Evaluation

In this section, we experimentally demonstrate the effectiveness of EdgeSanitizer using multiple real-world datasets and Apps. In addition, we implement the proposed framework on mobile phone and compare its performances to other solutions.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

A. Dataset and Configuration

To evaluate the performance of EdgeSanitizer, we consider a scenario where smartphone users want to train a motion-based activity classifier without revealing their data to others. We use the WISDM Human Activity Recognition dataset [33], which is a set of accelerometer data on an Android phone by 35 subjects performing 6 activities. Various time domain variables are extracted from the signal, and we consider the statistical measures obtained for every 10 seconds of accelerometer samples in [33] as $d = 43$ dimensional features in our models. Our final sample contains 5,418 accelerometer traces from 35 users, with 150.50 traces per user in average and a standard deviation of 44.73.

To obtain the ground-truth information for performance evaluation, some data is labeled for both the useful and sensitive inferences. The labeled training data is grouped to two different categories: mode-detection data and identity-recognition data. Users perform tasks such as walking, enunciating digits or specific alphabets, and the corresponding data segments are then labeled for the tasks. The mode-detection data corresponds to labeled user activities (e.g., accelerometer data segments are marked with labels such as walking), and speech-to-text translation labels (where audio segments are labeled with the corresponding spoken digit or alphabet). The identity-recognition data is used for authentication and speaker identity recognition experiments. The labeled data is generated by associating the identity of a user as label to a mobile device.

B. Evaluation Methodology

To evaluate the privacy of EdgeSanitizer, we provide provable privacy guarantees for temporal mobile data, which is segmented according to the parameter of time window size $N_w = 10$. For the deep learning based data minimization step, we use 10-fold cross validation to generate 90% of dataset as the training data and the remaining 10% as testing data. All the experiments are repeated for 1000 iterations, and the averaged results are reported. In our experiments, we used stacked autoencoders with two hidden layers comprising of 15 and 7 units respectively. We will show that an autoencoder with only two-hidden layers is able to extract better features than the state-of-the-art techniques. The reduced number of layers in the autoencoder allowed us to train the model using a small amount of labeled data from the user (note that we are protecting the sensitive inferences for a single user). We implement the case about the tradeoff between authentication and activity recognition on our real-world dataset using the system parameters discussed above.

C. Evaluation for Deep Inference Based on Data Minimization

To evaluate to the performance of EdgeSanitizer by privacy and utility guarantees, we conduct the deep inference based on data minimization model, where the useful inference is the behavior-based authentication. We further compare it with the existing feature extraction mechanisms: discrete Fourier transform (DFT) [34], discrete cosine transform (DCT) [35] and blind compressive sensing (BCS) [36].

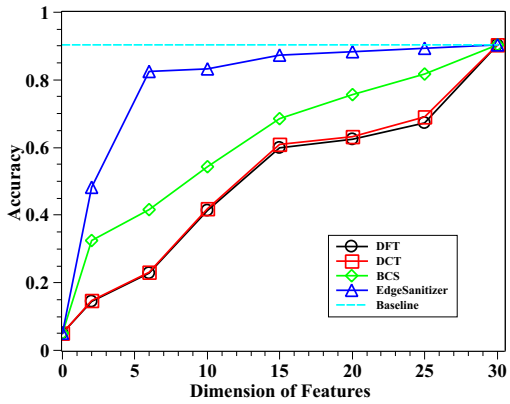


Fig. 3: The Utility of Different Feature Extraction Methods

Fig. 3 shows the utility-preserving under different feature extraction methods. Note that we use three-dimensional accelerometer measurement for behavior-based authentication and set the window size as $N_w = 10$, therefore the dimension of each segmented mobile data is 3×10. we have observe from Fig. 3 that: i) more features would help to improve the utility of data as the more features would be more accurate to represent the raw data. ii) EdgeSanitizer can achieve higher accuracy than several existing approaches with great improvement. 7 features are enough for EdgeSanitizer to provide utility with 87.27%accuracy, while the accuracy with all the 30 features is 90.45%. iii) Our proposed feature extraction approach greatly benefits from autoencoders which incorporate the minmax filters into deep inference models.

Based on our analysis above, we can find that edge servers have to provide incentives to mobile devices to contribute their data in order to reach a mutually satisfying trade-off between privacy and utility.

To show the scalability of EdgeSanitizer on numerous mobile devices, we briefly present an evaluation of EdgeSanitizer at edge servers by generating random networks with different sizes and densities.

For each generated random topology, we reported result that is the average of 100 executions. From the experimental result (as shown in Fig. 4), we can observe that EdgeSanitizer execution needs no more than 17s at edge servers, and a consensus can be achieved with different privacy budgets even in large networks with hundreds of mobile devices in less time.

D. Evaluation for Deep Inference with LDP

To demonstrate the effectiveness of EdgeSanitizer with LDP combining both feature extraction and perturbation, we again consider the behavior-based authentication as useful inference and the activity mode detection as sensitive inference. We compare the baseline method [37] and DP-ML [19] with our mechanism (i.e., Algorithm 1), since they achieve the same level of privacy guarantees for preventing all possible sensitive inferences.

Fig. 5 shows the utility performance computed over the obfuscated mobile data generated by EdgeSanitizer. We can make the following important observations using Fig. 5. i)

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

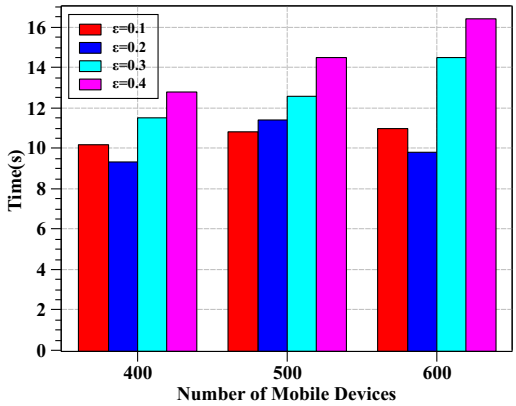


Fig. 4: Runtime Varying N and ϵ.

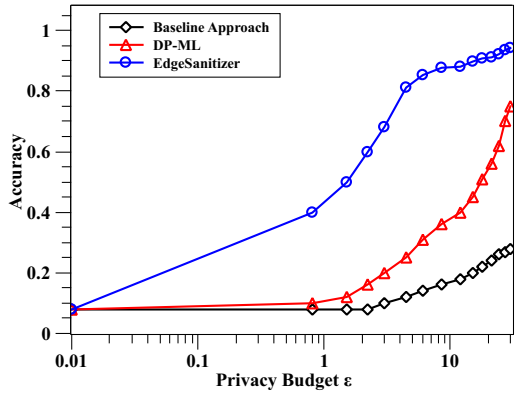


Fig. 5: The Utility of Deep Inference with LDP

EdgeSanitizer achieves considerable improvement over the baseline approach in terms of utility. This validates that EdgeSanitizer not only provides rigorous privacy guarantees for protecting sensitive inferences, but also retains the utility of the perturbed data. ii) EdgeSanitizer achieves better utility performance in the case which only considers specific sensitive inferences than that of the baseline approach which considers the entire set of sensitive inferences. iii) As expected, there is an improvement in utility at higher values of ε but at the cost of degradation in privacy. iv) Even at a moderate value of $\varepsilon = 5$ which is a typical privacy budget in LDP (similar values can also be found in Google's Rappor System [18]), the accuracy using EdgeSanitizer is close to the noise-free level. This observation further validates the effectiveness of our mechanisms. Note that the neighboring databases in LDP may differ in all their possible tuples (instead of differing in only one tuple as in DP). Thus, a proper privacy budget in LDP for balancing utility and privacy is usually higher than that of DP.

The false rejection rate (FRR) in Fig. 6(a) is the probability of desired features being misclassified as sensitive features, and the false acceptance rate (FAR) in Fig. 6(b) is the probability of sensitive features being misclassified as non-sensitive or desired features. In Fig. 6(a) and Fig. 6(b), we can observe that EdgeSanitizer under both usage modes achieve significantly smaller FRR and FAR compared to the baseline approach. In the sense, the accuracy of the sensitive inferences degrades at a much faster rate than that of the useful inferences when more noise is added (corresponding to a smaller privacy budget ε), which further validates the effectiveness of EdgeSanitizer.

To investigate the effectiveness of EdgeSanitizer on real-world mobile applications, we plot the trade-off between the accuracy of making useful inferences versus the accuracy of making sensitive inferences with the different number of features d, in Fig. 7. The results demonstrate that EdgeSanitizer works well in practice and returns an acceptable utility performance while satisfying provable privacy guarantees.

VII. Related Work

Previous privacy-preserving machine learning solutions adopt traditional perturbation mechanisms (e.g, randomized noise addition [38] and k-anonymity [39], [40]) against sensitive information inference. Some recent works are focusing on developing deep learning mechanisms with differential privacy to provide improved privacy guarantee while inferring useful information [19], [20], [41]. Thus, we review the related works about privacy-preserving deep learning from the following perspectives.

A. Privacy-preserving Machine Learning

Some prior works investigated the problem of privacy-preserving in machine learning from different angles. Some researchers try to eliminate the irrelevant data to increase the amount of uncertainty, while others try to protect data by cryptographic mechanisms. Additionally, some researchers focus on releasing datasets for some learning tasks [38], [42]. They usually concern about releasing a dataset includes high relevance features for learning tasks while not revealing individuals' privacy. Solutions such as k-anonymity [39], [40] and randomized noise addition [38] have been proposed. However, these solutions are only suitable for low-dimensional data.

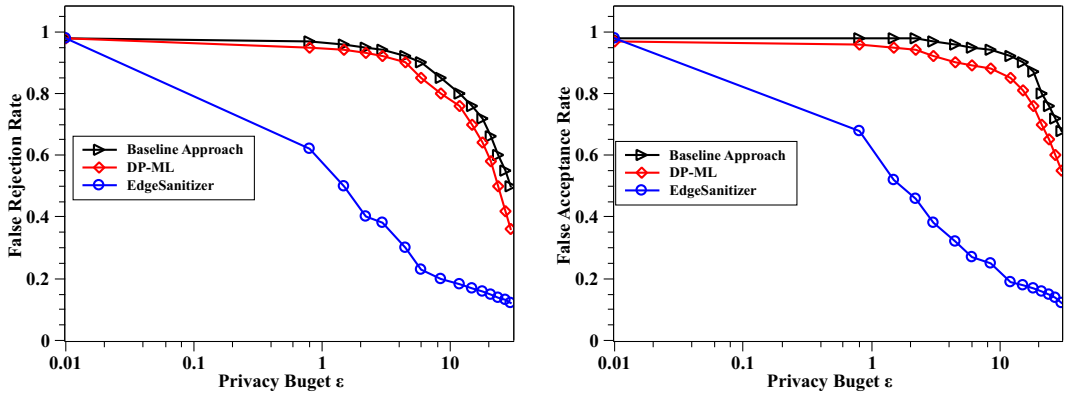
Additionally, distributed machine learning for privacy-preserving data inference and analytics is also a hot research topic. Several solutions [43]–[46] have been proposed for distributed privacy-preserving learning, where information is learnt from data owned by different features without disclosing the sensitive data. A different approach is proposed by Hamm *et al.* [44] and Papernot *et al.* [45], where privacy-preserving models are learned locally from disjoint datasets, and then combined on a privacy-preserving fashion. Liu *et al.* [46] design a collaborative deep learning mechanism with LDP for mobile application. The learn deep learning model is trained by multiple distributed data only by sharing partial parameters and keeping data in local to preserve privacy.

B. Deep Learning with Differential Privacy

Differential privacy [15] is an emerging method to release a dataset while keeping each individual record of the database private by injecting noise during the training. Thus, several differential privacy based deep learning solutions have been proposed recently to guarantee the confidentiality of personal data while inferring useful information [19], [20], [41]. Recently, Shokri *et al.* propose concern of privacy for deep

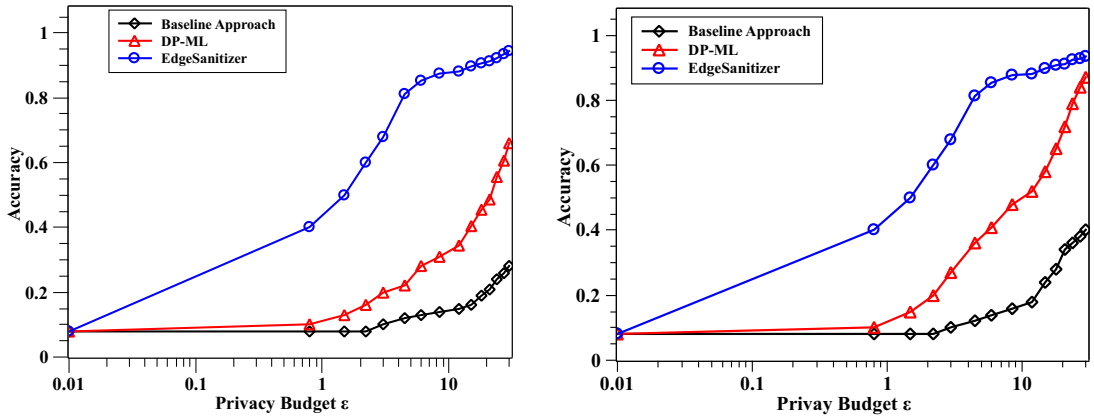
2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal



(a) False Rejection Rate (b) False Acceptance Rate

Fig. 6: False Rejection Rate and False Acceptance Rate with Varied Privacy Budget ε



(a) d = 7 (b) d = 30

Fig. 7: Trade-off between Utility and Privacy with Different Number of Features N_f

learning [20], and Abadi *et al.* present differential private stochastic gradient descent for deep learning (DP-ML) [19]. Shokri and Shmatikov adopt multiple parties collaboratively to learn a neural-network model for a specific tasks by sharing their learning parameters and not sharing their input datasets. Phan *et al.* [41] achieve ε-differential privacy by adding noise into objective functions of the deep autoencoders at every training step. In our scenario, user's data may not exist in training data and we focus on inference phase of a learning model.

Different from these solutions, our work focuses on design a new protection layer at the edge against sensitive inference. Moreover, we develop a novel technology to learn a deep inference by extraction of useful features from input mobile data, and obfuscate the learnt features by adaptively injecting random noise in terms of the contribution of each to the output.

VIII. Conclusion

Privacy preserving in mobile data analysis is a very challenging task, especially when the mobile data is gathered from different mobile devices. Privacy violation arises when personal data is used for sensitive inference by an adversary. In order to protect the data privacy against sensitive inference, in this paper, we have presented EdgeSanitizer, a deep inference framework based on edge computing with local differential privacy for mobile data analytics. EdgeSanitizer adopts deep learning based data minimization model to distill the undesired sensitive information with local differential privacy from mobile data. Then we have evaluated the performance of EdgeSanitizer by data privacy and utility using real-world Apps and datasets. The experimental results demonstrate that EdgeSanitizer can achieve high accuracy on the primary tasks, while heavily mitigating any sensitive inference potential for other tasks. Also by implementing the framework on a mobile phone, we show that EdgeSanitizer can greatly decrease the runtime for a specific task on the edge server, and provide provable privacy guarantees with a large improvement in utility.

The future work is to extend EdgeSanitizer by designing a framework for Learning as a Service (LAAS), where the users could share their data to train a new learning model with LDP. Another potential extension to our framework can

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

support other kinds of deep learning models such as deep reinforcement learning and also other applications for speech or video processing.

ACKNOWLEDGMENT

This work is supported by the Young and Middle-Aged Scientific Research Project in the Department of Education of Fujian (No. JAT160469/B201618), Natural Science Foundation of China (Nos. 61702562 and 61702561), 111 Project (No. B18059), Scientific Research Fund of Hunan Education Department (No. 17C0866), Innovation-Driven Project of Central South University (No. 2016CXS013), International Science & and Technology Cooperation Program of China (No. 2013DFB10070) and China Hunan Provincial Science & Technology Program (No. 2012GK4106). Kui's research is supported in part by US National Science Foundation under grant CNS-1262277.

REFERENCES

[1] Y. Zhang, J. Ren, J. Liu, C. Xu, H. Guo, and Y. Liu, "A survey on emerging computing paradigms for big data," *Chinese Journal of Electronics*, vol. 26, no. 1, pp. 1–12, 2017.

[2] X. Peng, J. Ren, L. She, D. Zhang, J. Li, and Y. Zhang, "Boat: A block-streaming app execution scheme for lightweight iot devices," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1816–1829, 2018.

[3] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable iot architecture based on transparent computing," *IEEE Network*, vol. 31, no. 5, pp. 96–105, 2017.

[4] H. Guo, J. Liu, J. Zhang, W. Sun, and N. Kato, "Mobile-edge computation offloading for ultra-dense iot networks," *IEEE Internet of Things Journal*, 2018.

[5] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Mobile edge computing and networking for green and low-latency internet of things," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 39–45, 2018.

[6] F. Tang, Z. M. Fadlullah, B. Mao, and N. Kato, "An intelligent traffic load prediction based adaptive channel assignment algorithm in sdn-iot: A deep learning approach," *IEEE Internet of Things Journal*, 2018.

[7] K. Zhang, Y. Mao, S. Leng, Y. He, and Y. Zhang, "Mobile-edge computing for vehicular networks: A promising network paradigm with predictive off-loading," *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 36–44, 2017.

[8] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Cooperative content caching in 5g networks with mobile edge computing," *IEEE Wireless Communications*, vol. 25, no. 3, pp. 80–87, 2018.

[9] L. Li, K. Ota, and M. Dong, "Deep learning for smart industry: Efficient manufacture inspection system with fog computing," *IEEE Transactions on Industrial Informatics*, 2018.

[10] S. Nirjon, R. F. Dickerson, P. Asare, Q. Li, D. Hong, J. A. Stankovic, P. Hu, G. Shen, and X. Jiang, "Auditeur: A mobile-cloud service platform for acoustic event detection on smartphones," in *Proceeding of the 11th annual international conference on Mobile systems, applications, and services(MobiSys)*, 2013, pp. 403–416.

[11] C. Xu, J. Ren, D. Zhang, and Y. Zhang, "Distilling at the edge: A local differential privacy obfuscation framework for iot data analytics," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 20–25, 2018.

[12] J. Wu, M. Dong, K. Ota, J. Li, and Z. Guan, "Fcss: Fog computing based content-aware filtering for security services in information centric social networks," *IEEE Transactions on Emerging Topics in Computing*, 2017.

[13] L. Pournajaf, D. A. Garcia-Ulloa, L. Xiong, and V. Sunderam, "Participant privacy in mobile crowd sensing task management: A survey of methods and challenges," *ACM SIGMOD Record*, vol. 44, no. 4, pp. 23–34, 2016.

[14] W. Tang, K. Zhang, J. Ren, Y. Zhang, and X. S. Shen, "Flexible and efficient authenticated key agreement scheme for bans based on physiological features," *IEEE Transactions on Mobile Computing*, 2018, doi: 10.1109/TMC.2018.2848644.

[15] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 2006, pp. 486–503.

[16] C. Xu, J. Ren, Y. Zhang, Z. Qin, and K. Ren, "Dppro: Differentially private high-dimensional data release via random projection," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 12, pp. 3081–3093, 2017.

[17] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, 2013, pp. 429–438.

[18] Ú. Erlingsson, V. Pihur, and A. Korolova, "Rappor: Randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*. ACM, 2014, pp. 1054–1067.

[19] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 308–318.

[20] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, 2015, pp. 1310–1321.

[21] H. Li, K. Ota, and M. Dong, "Learning iot in edge: deep learning for the internet of things with edge computing," *IEEE Network*, vol. 32, no. 1, pp. 96–101, 2018.

[22] F. D. McSherry, "Privacy integrated queries: an extensible platform for privacy-preserving data analysis," in *Proc. of ACM SIGMOD 2009*, 2009, pp. 19–30.

[23] N. Saleheen, S. Chakraborty, N. Ali, M. M. Rahman, S. M. Hossain, R. Bari, E. Buder, M. Srivastava, and S. Kumar, "msieve: differential behavioral privacy in time series of mobile sensor data," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2016, pp. 706–717.

[24] J. Ren, Y. Zhang, R. Deng, N. Zhang, D. Zhang, and X. Shen, "Joint channel access and sampling rate control in energy harvesting cognitive radio sensor networks," *IEEE Transactions on Emerging Topics in Computing*, 2016, doi: 10.1109/TETC.2016.2555806.

[25] W.-H. Lee and R. B. Lee, "Multi-sensor authentication to improve smartphone security," in *Information Systems Security and Privacy (ICISSP), 2015 International Conference on*, 2015, pp. 1–11.

[26] Y. Nam, Y. Kim, and J. Lee, "Sleep monitoring based on a tri-axial accelerometer and a pressure sensor," *Sensors*, vol. 16, no. 5, p. 750, 2016.

[27] M. Mehrnezhad, E. Toreini, S. F. Shahandashti, and F. Hao, "Touchsignatures: identification of user touch actions and pins based on mobile sensor data via javascript," *Journal of Information Security and Applications*, vol. 26, pp. 23–38, 2016.

[28] S. Li, L. Da Xu, and X. Wang, "Compressed sensing signal and data acquisition in wireless sensor networks and internet of things," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2177–2186, 2013.

[29] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[30] J. Zhang, Z. Zhang, X. Xiao, Y. Yang, and M. Winslett, "Functional mechanism: regression analysis under differential privacy," *Proceedings of the VLDB Endowment*, vol. 5, no. 11, pp. 1364–1375, 2012.

[31] J. Hamm, "Enhancing utility and privacy with noisy minimax filters," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 6389–6393.

[32] T. M. Apostol, *Calculus*. John Wiley & Sons, 1967.

[33] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 2, pp. 74–82, 2011.

[34] S. Weinstein and P. Ebert, "Data transmission by frequency-division multiplexing using the discrete fourier transform," *IEEE transactions on Communication Technology*, vol. 19, no. 5, pp. 628–634, 1971.

[35] K. R. Rao and P. Yip, *Discrete cosine transform: algorithms, advantages, applications*. Academic press, 2014.

[36] S. Gleichman and Y. C. Eldar, "Blind compressed sensing," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6958–6975, 2011.

[37] C. Liu, S. Chakraborty, and P. Mittal, "Deeprotect: Enabling inference-based access control on mobile sensing applications," *arXiv preprint arXiv:1702.06159*, 2017.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2897005, IEEE Internet of Things Journal

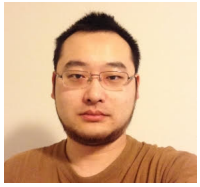
[38] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 2001, pp. 247–255.
[39] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Incognito: Efficient full-domain k-anonymity," in *Proc. of ACM SIGMOD 2005*, 2005, pp. 49–60.
[40] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, 2007, pp. 106–115.
[41] N. Phan, Y. Wang, X. Wu, and D. Dou, "Differential privacy preservation for deep auto-encoders: an application of human behavior prediction." in *AAAI*, 2016, pp. 1309–1316.
[42] V. S. Iyengar, "Transforming data to satisfy privacy constraints," in *Proc. of ACM SIGKDD 2002*, 2002, pp. 279–288.
[43] M. Li, D. G. Andersen, A. J. Smola, and K. Yu, "Communication efficient distributed machine learning with the parameter server," in *Advances in Neural Information Processing Systems*, 2014, pp. 19–27.
[44] J. Hamm, Y. Cao, and M. Belkin, "Learning privately from multiparty data," in *International Conference on Machine Learning*, 2016, pp. 555–563.
[45] N. Papernot, M. Abadi, Ú. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," in *the 5th International Conference on Learning Representations*, 2017.
[46] M. Liu, H. Jiang, J. Chen, A. Badokhon, X. Wei, and M.-C. Huang, "A collaborative privacy-preserving deep learning system in distributed mobile environment," in *Computational Science and Computational Intelligence (CSCI), 2016 International Conference on*. IEEE, 2016, pp. 192–197.

Chugui Xu [S'16] (chuguixu@csu.edu.cn) received the Ph.D degree in Computer Science (2018) from Central South University, China. He has received the B.S. degree in Physics (2005) from Hunan Normal University, China. He also holds the M.S. degree in Computer Science (2010) from Hunan University of Technology, China. His research interests include Internet of Things, data privacy and security.

Ju Ren [S'13, M'16] (renju@csu.edu.cn) received the B.Sc. (2009), M.Sc. (2012), Ph.D. (2016) degrees all in computer science, from Central South University, China. During 2013-2015, he was a visiting Ph.D. student in the Department of Electrical and Computer Engineering, University of Waterloo, Canada. Currently, he is a professor with the School of Computer Science and Engineering, Central South University, China. His research interests include Internet-of-Things, wireless communication, network computing and cloud computing. He is a co-recipient of the best paper award of IEEE IoP 2018 and the most popular paper award (2015-2018) of Chinese Journal of Electronics. He currently serves/served as an associate editor for IEEE Transactions on Vehicular Technology and Peer-to-Peer Networking and Applications, and a TPC member of many international conferences including IEEE INFOCOM19/18, Globecom17, WCNC17, WCSP16, etc. He also served as a poster co-chair of IEEE MASS18, a track co-chair for IEEE VTC17 Fall and IEEE I-SPAN18, and an active reviewer for over 20 international journals. He is a member of IEEE and ACM.

Liang She [S'17] (sheliang@csu.edu.cn) is a lecturer with the School of Computer and Information Engineering, Hunan University of Commerce, China. He received his B.S. and M.S. degrees in computer science from Central South University, China, in 2002 and 2011, respectively. He is pursuing a Ph.D. in computer science and technology at the School of Computer Science and Engineering from Central South University. His research interests include Internet of Things, Data Mining and Personalized Recommender Systems. He is a member of ACM and CCF.

Yaoxue Zhang [M'17, SM'18](zyx@csu.edu.cn) received his B.Sc. degree from Northwest Institute of Telecommunication Engineering, China, in 1982, and his Ph.D. degree in computer networking from Tohoku University, Japan, in 1989. Currently, he is a professor with the School of Computer Science and Engineering, Central South University, China, and also a professor with the Department of Computer Science and Technology, Tsinghua University, China. His research interests include computer networking, operating systems, ubiquitous/pervasive computing, transparent computing, and big data. He has published over 200 technical papers in international journals and conferences, as well as 9 monographs and text-books. Currently, he is serving as the Editor-in-Chief of Chinese Journal of Electronics. He is a fellow of the Chinese Academy of Engineering.



Zhan Qin (zhanqin@buffalo.edu) is a professor in Institute of Cyberspace Research of Zhejiang University. He received his Ph.D. degree at the director of the Ubiquitous Security and Privacy Research Laboratory (UbiSeC) in the Computer Science and Engineering Department of the State University of New York at Buffalo, NY, USA. His research interests focus on Data Privacy, Crowdsourcing Security and Smart Grid.

Kui Ren [F'16] (kuiren@buffalo.edu) is a professor in Institute of Cyberspace Research of Zhejiang University and the director of UbiSeC Lab at State University of New York at Buffalo (UB). He received his PhD degree from Worcester Polytechnic Institute. Kui's current research interest spans Cloud & Outsourcing Security, Wireless & Wearable Systems Security, and Mobile Sensing & Crowdsourcing. Kui has published 200 papers in peer-reviewed journals and conferences and received several Best Paper Awards, including IEEE ICDCS 2017, IWQoS 2017, and ICNP 2011. He received IEEE CISTC Technical Recognition Award in 2017, UB Exceptional Scholar Award for Sustained Achievement in 2016, UB SEAS Senior Researcher of the Year Award in 2015, Sigma Xi/IIT Research Excellence Award in 2012, and NSF CAREER Award in 2011. He currently serves on the editorial boards of IEEE Trans. on Dependable and Secure Computing, IEEE Trans. on Service Computing, IEEE Trans. on Mobile Computing, IEEE Wireless Communications, IEEE Internet of Things Journal, and SpingerBriefs on Cyber Security Systems and Networks. Kui is a Fellow of IEEE, a Distinguished Lecturer of IEEE, a member of ACM, and a past board member of Internet Privacy Task Force, State of Illinois.

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.