IFed: A novel federated learning framework for local differential privacy in Power Internet of Things

Hui Cao¹, Shubo Liu¹, Renfang Zhao² and Xingxing Xiong¹

Abstract

Nowadays, wireless sensor network technology is being increasingly popular which is applied to a wide range of Internet of Things. Especially, Power Internet of Things is an important and rapidly growing section in Internet of Thing systems, which benefited from the application of wireless sensor networks to achieve fine-grained information collection. Meanwhile, the privacy risk is gradually exposed, which is the widespread concern for electricity power consumers. Non-intrusive load monitoring, in particular, is a technique to recover state of appliances from only the energy consumption data, which enables adversary inferring the behavior privacy of residents. There can be no doubt that applying local differential privacy to achieve privacy preserving in the local setting is more trustworthy than centralized approach for electricity customers. Although it is hard to control the risk and achieve the trade-off between privacy and utility by traditional local differential privacy obfuscation mechanisms, some existing obfuscation mechanisms based on artificial intelligence, called advanced obfuscation mechanisms, can achieve it. However, the large computing resource consumption to train the machine learning model is not affordable for most Power Internet of Thing terminal. In this article, to solve this problem, IFed was proposed—a novel federated learning framework that let electric provider who normally is adequate in computing resources to help Power Internet of Thing users. First, the optimized framework was proposed in which the trade-off between local differential privacy, data utility, and resource consumption was incorporated. Concurrently, the following problem of privacy preserving on the machine learning model transport between electricity provider and customers was noted and resolved. Last, users were categorized based on different levels of privacy requirements, and stronger privacy guarantee was provided for sensitive users. The formal local differential privacy analysis and the experiments demonstrated that IFed can fulfill the privacy requirements for Power Internet of Thing users.

Keywords

Local differential privacy, differential privacy, federated learning, IoT, Power Internet of Things

Date received: 23 January 2020; accepted: 17 March 2020

Handling Editor: Xiaojiang Du

Introduction

Recent years have witnessed the great growth of Internet of Things (IoTs), which have developed significantly in various fields,¹,²,³ especially in Power IoTs. As a smart electric power system that realizes the interconnection of all things in the grid, human-computer interaction, and comprehensive state perception, the Power IoTs are applied to modern information, such as mobile internet and artificial intelligence, around all aspects of the power system. In particular, different from core power transmission system, Power IoTs installed a large number of smart

¹School of Computer Science, Wuhan University, Wuhan, China
²North China Branch of State Grid Corporation of China, Beijing, China

Corresponding author:
Shubo Liu, School of Computer Science, Wuhan University, Wuhan 430072, China.
Email: liu.shubo@whu.edu.cn
meters and sensors for fine-grained information collection. Wireless sensor networks (WSNs) are promising option on Power IoT systems, which are provided with low cost and large geographic area coverage. As shown in Figure 1, besides the strong electric power transmission channel, in home area networks (HANs) and neighbor area networks (NANs), power grid system needs a broad coverage of data transmission channel in which WSNs are applicable.

IoT technologies have the capability to collect, quantify, and understand the surrounding environment, which brings many benefits to IoT users. However, the extensive users’ data collection and processing from IoT device, such as smart meter, also bring some privacy concerns. As the IoT end-devices can be deeply involved in users’ private data, the data generated by them will contain privacy-sensitive information. Collected data in the IoT device may leak and threaten smart grid users’ behavior privacy. For example, by applying non-intrusive load monitoring (NILM) techniques, power consumption data could infer users’ privacy. In Figure 2, it is possible to infer that when the fan heater, stove burner, or other electric appliances are in use, this may infer the detailed behavior of the residents, even might infer the identity or personal privacy of the residents continually. These will be collectively referred to as the behavior privacy roughly, which will be mainly exposed in this article.

Differential privacy (DP), which has been used successfully in various fields, provided a formalization of the notion of a privacy adversary, the introduction to a meaningful measure of privacy loss. In traditionally centralized DP, privacy is guaranteed by adding obfuscation to output of trusted data aggregator. However, Power IoT networks are the network systems which consist of massive smart meters, sensors, or other IoT devices, embedded in the physical world widespread with weak network boundary. Adversary may conceal from anywhere on the smart grid user zone. There are many potential attacks, such as NILM, that can reveal the private data before they reach the trusted data curator. Furthermore, electric power grid users might be sensitive military industrial enterprises, and electric power provider cannot be regarded as trusted third party, moreover the widely uncontrollable channels. In summary, in Power IoTs, the support of trusted data curator is inadequate, therefore local obfuscation is a better choice for users’ behavior privacy preserving in Power IoTs.

Local differential privacy (LDP) has been used in privacy preservation of smart grid and IoT in recent years. LDP avoids collecting exact original power consumption information and substitutes it with locally disturbed data onto user side, thus providing a stronger assurance to the users. Unfortunately, because of the robustness of NILM, traditional obfuscation mechanisms, such as randomized rescannot reduce the accuracy of behavior inference observably. However, for accurate billing, energy consumption data are high accuracy sensitive and excessive obfuscation is not acceptable to customers and electricity provider. It is deemed hard to achieve the trade-off between user behavior privacy and utility of energy consumption.

Figure 1. Architecture for Power Internet of Things. The black solid line is the electric power flow and the blue solid line is the information flow.
data for ordinary obfuscation mechanisms. Many researches had indicated that naive hardware implementation of LDP mechanism cannot guarantee behavior privacy of individual user in Power IoT systems. Some advanced obfuscation mechanisms combined with distribution estimation or machine learning are able to achieve the trade-off; however, the following problem is that the naive Power IoT device cannot endure the complex and high computational cost algorithms, especially the procedure of model training that high-computing resource consumption. In summary, Power IoTs with naive hardware cannot guarantee behavior privacy completely through directly applying existing local obfuscation mechanisms.

To solve the behavior privacy in Power IoTs, we proposed IFed, a novel federated learning framework of IoT. As an extension of federated learning, IFed uses model transport instead of sensitive data transport to privacy preserving. The main focus of this article is to derive insights into the trade-off between behavior privacy of power internet of things users and utility of energy consumption data, more importantly, grapple with the problem of how does complex advanced obfuscation mechanisms suitable for naive Power IoT user side with very low computation capacity. We aim to achieve a good trade-off between users’ behavior privacy against NILM, data utility, and low computational cost at Power IoT device. Besides, we strictly and formally define and prove its privacy protection strength.

The contributions of the article are listed as follows:

- To the best of our knowledge, this is the first work that federated learning framework for the IoT systems, which could solve the users’ behavior privacy problem against NILM technology and achieve the trade-off between privacy and utility;
- Especially, our solution solves the behavior privacy problem of not only the process of energy consumption data uploads but also the consequent process of model training that is likely to be a potential behavior privacy risk in the procedure that models uploads and downloads between users and electric provider;
- Considering existing conditions in smart grid that many sensitive users and regular users exist together, our solution supports different privacy requirements of users and different obfuscation mechanisms.

The remainder of this article is structured as follows. Section “Preliminaries” provides background on NILM, LDP, and obfuscation mechanisms. Section “System model” proposes the overview of IFed. Section “Key algorithms in IFed” discusses the three key algorithms used in IFed in detail—model aggregation, horizontal federated learning (HFL), and heterogeneous federated transfer learning (HFTL). Section “LDP analysis” explains the formal analysis by LDP. Section “Experiments” describes the evaluation and experimental results. Finally, section “Conclusion” gives the conclusion.

Preliminaries

In this section, the adversary model, LDP, and obfuscation mechanisms are introduced which are used for federated learning for LDP in IoTs.

Adversary model

Adversary may be hidden and snooped everywhere on the wide grid, even around the resident. The threat stems from attacker inferring users’ behavior and action with high confidence from energy consumption observers. This technology called non-intrusive load monitoring was first introduced by Hart. In recent years, some research works improved NILM using artificial intelligence algorithms and with measurements at higher accuracy. In particular, some improved NILM algorithms have had favorable robustness, which can keep high accuracy under obfuscation. In the case of single load identification, it compares the extracted feature of unknown loads with those of known loads in the device database pool and tries to minimize the errors between them to find the closest match.
of the meter at \( t = 1, 2, \ldots, T \), the task of the NILM algorithm is to infer the power contribution \( y_i^t \) of appliance \( i \in 1, 2, \ldots, T \) at time \( t \), such that at any point in time \( t \)

\[
X^t = \sum_{i=1}^{N} y_i^t + \sigma(t)
\]

where \( \sigma(t) \) represents any contribution from appliances not accounted for noise measurement. The object of NILM is

\[
\text{class} = \arg\min \| \hat{y}_i - y_i \|
\]

where \( \hat{y}_i \) is the appliance feature available in the signature library and \( y_i \) is the new feature extracted due to occurrence of an unknown event.

**LDP**

Traditional global differential privacy (GDP)\(^9\) and LDP\(^23\) are two approaches to achieve DP. Different from GDP, LDP needs no trusted data curator,\(^33\) which is more applicable to behavior privacy in Power IoTs. Recently, an increasing number of researchers had changed their focus to LDP in IoTs, since LDP can enable protecting each user’s privacy locally without relying on a trusted third party.\(^34\) LDP can be formulated as follows.

**Definition 1 (\( \varepsilon \)-LDP).** An obfuscation mechanism \( M \) satisfies \( \varepsilon \)-LDP if and only if for any pairs of adjacent input tuples \( x \) and \( x' \) in the domain of \( M \), and for any possible subset of outputs \( Z \), it always holds

\[
P[M(x) \in z] \leq e^\varepsilon \cdot P[M(x') \in z]
\]

where \( P \) denotes probability.

Thus it can be seen that the obfuscation mechanism plays a key role in LDP; most of the existing obfuscation mechanisms are random and are based on randomized response such as RAPPOR.\(^23\) The randomized response can be formulated as follows.

**Randomized response.** For each client’s value \( v \) and bit \( i, 0 \leq i \leq k \) in \( B \), create a binary reporting value \( B_i^t \) which equals to

\[
B_i^t = \begin{cases} 
1, & \text{with probability } \frac{1}{2}f \\
0, & \text{with probability } \frac{1}{2}f \\
B_i, & \text{with probability } 1-f
\end{cases}
\]

where \( f \) is a parameter controlling the level of longitudinal privacy guarantee. Then, send the generated report \( B_i^t \) to the server.

**Advanced obfuscation mechanisms**

We find that some mechanisms proposed recently are no longer completely randomly obfuscated but transformed first based on the state of the target before applying LDP rather than the traditional method; this kind of mechanism is more efficient to achieve the trade-off between utility and privacy in IoTs. They can be split into several categories which will be introduced in this section; however, we call them as advanced obfuscation mechanisms.

One is based on discrete distribution estimation,\(^24,25\) where before randomized response, empirical estimation is carried out first as

\[
m = \frac{e^\varepsilon - 1}{e^\varepsilon + k - 1} \hat{p} + \frac{1}{e^\varepsilon + k - 1}
\]

where \( m \) is the output of empirical estimation, and the empirical estimation of \( p \) is given by

\[
\hat{p} = \frac{e^\varepsilon + k - 1}{e^\varepsilon - 1} \hat{m} - \frac{1}{e^\varepsilon - 1}
\]

where \( \hat{m} \) is the empirical estimate of \( m \)

\[
\hat{m} = \frac{1}{e^\varepsilon - 1} \begin{cases} 
e e^\varepsilon + k - 2 & \text{if } y = x \\
-1 & \text{if } y \neq x
\end{cases}
\]

Another one is based on machine learning such as Bayes,\(^35\) cluster,\(^26\) Markov,\(^27\) and sparse coding.\(^28\) Usually, this kind of mechanism first performs an algorithm before randomized response. For example, in sparse coding, a dictionary is trained by objective function

\[
\| Y_{t \times 1} - B_{t \times n} A_{n \times 1} \| + \lambda f(A_{n \times 1})
\]

where \( \lambda f(A_{n \times 1}) \) is the restrictive condition of activation matrix \( A_{n \times 1} \). The dictionary is

\[
\hat{A} = \arg\min \| Y_{t \times 1} - [b_1 \ldots b_n \begin{bmatrix} a_1 \\ \vdots \\ a_i \end{bmatrix}] \| + \lambda \sum_{i} ||A||
\]

The key problem of the advanced obfuscation mechanisms in Power IoTs, at the same is the major challenge of this article, is computing resources that are in great demand. Most of the Power IoTs, probably smart meters, are unrealistic.

**Federated learning**

The concept of federated learning\(^29,36\) is proposed by Google recently to reduce the risk of the cloud service provider learning the personal model update. In federated learning, model is learned by multiple clients in decentralized fashion.\(^37,38\) The parameters of trained
model are centralized by a trusted curator, and then it distributes an aggregated model back to the clients.  

In particular, the goal is typically to minimize the following objective function

$$\arg \min_w \sum_{k=1}^m p_k F_k(w)$$

where $m$ is the number of clients, $F_k$ is the local objective function for the $k$th client, and $p_k$ specifies the relative impact of each client.

**System model**

In this section, we present the design rationale of the federated framework for LDP in Power IoTs.

**Categorization of IoT user**

Generally speaking, Power IoT is a wide network involving many sections such as HAN, NAN, and WAN, and the system structure is quite different between different regions around the world. Meanwhile, the requirements and investments of privacy preserving for different users are not completely the same, therefore the requirement of privacy protection for different users may be different.

According to the privacy of press and investment, we categorized IoT users as sensitive users and regular users. Our aim is to provide a stronger privacy guarantee for sensitive users, meanwhile making full use of the local edge computing equipment of sensitive users. Regular users, such as residents, usually seek a good privacy without high investment. Their behavior privacy disclosure may lead to personal privacy loss. We assume that such users do not want to send the original data that may lead to privacy disclosure directly; however, they can accept the use of models from grid and send local models to grid, as they may not be willing to add any devices except smart meters.

Sensitive users, such as military industry enterprises, may seek an absolute privacy with some acceptable investments. Their behavior privacy disclosure may lead to the disclosure of top secrets or great loss. We assume that such users do not trust other users in the grid, data transmission channel, or grid side. And sensitive users do not accept the un-obfuscated original energy consumption data sent to others for billing or for model training; however, they accept it by adding some local IoT devices with weak computing resources to support the privacy protection.

Due to the low computation capability of IoT devices, sensitive users have no capability to perform complicated computational tasks such as fully deep learning model training. However, they have the ability to perform some simple tasks such as model execute to prediction to HFTL. Furthermore, regular users only have the ability to perform some more simple tasks such as model execute to prediction to homogeneous transfer learning.

**Overview of Ifed**

The aim at our framework is letting electricity provider help users to train model; however, it does not mean electricity provider had to be seen as a completely trusted third party and the users give up privacy protection locally. In fact, our goal is designing a system in such a way that the user can safely use trained model with the aid of incompletely trusted electricity provider, moreover defeat the adversary around the user. The expected result is that the adversary knows that the user transfers the consumption data with model or distribution estimation but they cannot recover it, meanwhile the data of high utility can be used in billing. Detailed goals are as follows:

- First, achieve the trade-off between behavior privacy against NILM, data utility–supported electricity bill, and low computing resource consumption in naive Power IoT terminal;
- Second, all models transport between users and electricity provider subjected to LDP, which will not be a new privacy risk;
- Note that the proposed federated learning for IoTs does not intend to replace the existing advanced obfuscation mechanisms but aims to solve the problem that power consumption data and trained model safely interact between individuals and grid side.

The architecture of the federated learning system in Power IoTs is shown in Figure 3. In this system, r regular users with similar data structure and u sensitive users with different data structure and different electric appliances coexist on same Power IoTs. To solve the problem caused by insufficient resources on users, electricity provider pre-trained a global model $w^g$ or tentatively undertake others’ task to substitute for users $k$ and $u$ individually. Then, the models $w'_1$, $w'_2$, ..., $w'_r$ and $w'^g_1$, $w'^g_2$, ..., $w'^g_u$ were fine tuned with transfer learning by local cached historical data as a new local model, and the power consummation data were obfuscated. In order to maintain stronger applicability and generality of global model, regular user uploads local model with LDP to electricity provider, by contrast, sensitive users do not participate in it for more absolute privacy guarantee. Then, the electric provider aggregates the model $w'_1$, $w'_2$, ..., $w'_r$ for a new global model $w'^g$. Sensitive user needs federated transfer learning technology to implicate global model to local model. The detailed process
of such a system usually contains the following five steps:

**Step 1.** Power provider send current pre-trained global model $w^0_g$ to all sensitive users $u$ and regular users $r$.

**Step 2.** Regular users $r_1, r_2, \ldots, r_t$ update local models $w^1_r, w^2_r, \ldots, w^t_r$ with global model $w^0_g$ and individual local cached data by HFL (see section “HFL for regular users”).

**Step 3.** Sensitive users $u_1, u_2, \ldots, u_n$ update local models $w^1_u, w^2_u, \ldots, w^t_u$ with global model $w^0_g$ and individual local cached data by HFTL (see section “HFTL for sensitive users”).

**Step 4.** Only regular users $r_1, r_2, \ldots, r_t$ generate the new local models obfuscated as $w^t_{r_1}, w^t_{r_2}, \ldots, w^t_r$ and then upload it to power provider.

**Step 5.** Power provider selects some regular users as active users $a_1, a_2, \ldots, a_t$ and generate a new global model by model aggregation (see section “Model aggregation for provider”).

### Key algorithms in IFed

In this section, the details of several key technology designs of this work are presented.

#### Model aggregation for provider

First, in this section, we introduce how the electricity provider learns a shared model by aggregating locally computed updates. As traditional federated learning, we select a $C$-fraction, which controls the global batch size, and a fixed learning rate $\eta$. Let $w^l_g$ denote the global model and $w^l_{g_{t+1}}$ denote the next round global model to learn. Then, the electricity provider aggregates these gradients and applies the update $w^l_g$ as

$$w^{l+1}_g \leftarrow w^l_g - \eta \sum_{k=1}^{K} \frac{n_k}{R} g_k$$

where $g_k = \nabla f_k(w^l)$.

Typically, we take $f_k(w) = l(x_i, y_i; w)$ as the loss of the prediction on example $(x_i, y_i)$ under model $w$, and in this work, we adopt it as cross-entropy loss. Because the number of regular users may be very large, we cannot update the global model with all their local model, which is inefficient and not necessary. Optimizing it as random select $C$-fraction in all regular users, we have

$$K = CR$$

$$\sum_{k=1}^{K} \frac{n_k}{R} g_k = \nabla f(w^l)$$

where $C$ is the $C$-fraction and $R$ is the number of regular users.

In summary, our model aggregation is similar to federated stochastic gradient descent (FedSGD); however, we can apply it to not only the algorithms based on neural network but also other advanced mechanisms such as those based on Markov or sparse coding like...
Algorithm 1. Federated learning for local differential privacy algorithm. The R regular users are randomly selected by C-fraction and \( \eta \) is the learning rate.

**[Grid Side Execution]:**
1. **Input:** \( u \) is sensitive users, \( r \) is regular users, and \( I \) is the number of smart meter random select for activation
2. for each round \( t = 0, 1, \ldots \) do
3. \( K = CR \)
4. for each regular activated user \( Kdo \)
5. \( w_k^t = w_t - \eta g_k \)
6. end for
7. \( w_{i+1} = \sum_k^{K} n_k w_k^t \)
8. end for

\[ w_{i+1} = w_{i} - \eta g_k \]

We will introduce this in detail in the following section. Then, \( w_{i+1} = \sum_k^{K} n_k w_k^t \). In summary, the work for regular user is given by Algorithm 1.

**HFL for regular users**

Consider that regular users \( R \), in most cases, are common residents or enterprises with similar live or business behavior, therefore we assume the feature spaces are the same

\[ X_{R_i} = X_{R_j}, Y_{R_i} = Y_{R_j} \]

where \( X \) denotes the feature space and \( Y \) denotes the label space.

However, every user, such as smart meter, is unique. Therefore, data held by each data owner \( i \) and their space denoted by \( I \) are

\[ I_i \neq I_j, \forall D_i, D_j, i \neq j \]

**Learning procedure.** In this section, we will introduce the learning procedure of regular users, specially, how to generate local model from a global model and local cached data. First, the regular smart meter downloads the global model from electric power provider and overwrites local parameters. Then, he runs one epoch of SGD training on his local dataset when using obfuscation mechanism based on neural network. He runs one epoch dictionary update in sparse coding or similar performance in other obfuscation mechanisms. Third, all smart meters \( i, i \in R \), compute \( \Delta w^i \) that reflect how much each parameter has to change to more accurately model the local dataset of the \( i \)th smart meter

\[ \Delta w^i = \nabla w^i = \sum_{k=1}^{K} w_{j+k}^t - w_{j}^t \]

Note that performing one epoch of SGD training or one round of dictionary update is a much lower than full training. It is a simple task, which smart meter can complete.

**Objective function.** While we focus on regular users’ objectives, the algorithm we consider is applicable to minimize the loss between local cached data and local parameters.

In neural network, the objective function is

\[ \arg \min_{\Theta} L = \sum_{i=1}^{n} l(y_i^a, w_i(x_i^a)) \]

where \( l(a, b) \) denotes the loss function between \( a \) and \( b \), in this work we adopt it as cross-entropy loss; and \( \Theta \) denotes all the parameters to be learned.

In sparse coding, the objective function is

\[ \arg \min_{\Theta} L = \| x^a - [B_1 \ldots B_n] \begin{bmatrix} A_1 \\ \vdots \\ A_n \end{bmatrix} \| \]

**Obfuscation and upload.** To avoid recovering the privacy data onto local model by adversary, regular users could choose to obfuscate the local model before uploading it. One approach to obfuscation is that all regular users could use a fixed Gaussian mechanism like

\[ \mathcal{N}(0, \sigma^2) \]

where \( \mathcal{N} \) is a standard normal distribution and \( \sigma \) is its variance.

Regular smart meter uploads the local model as

\[ w^i = w^i + \mathcal{N}(0, \sigma^2) \]

Another approach is that each regular user uses a different Gaussian mechanism according to his model as

\[ \mathcal{N}(0, \sigma^2 S_f) \]

where \( S_f \) is the sensitivity of local model.

In summary, all work for regular user is given in Algorithm 2.

**HFTL for sensitive users**

Now we consider that regular users \( S \), a more complex case, are common sensitive electric power customers, who have got some specialty electric appliances which nobody else has received. The state of these appliances maybe the major concern of behavior privacy protection.
Algorithm 2. Federated learning for regular smart grid or IoT users.

R is the number of regular users (smart meter), the R regular users are randomly selected by C-fraction, K is the number of smart meter random select for activation, and \( \eta \) is the learning rate.

**Input:** pre-trained model \( w^f \) from the grid, a small amount of local history data \( D_{r(i)} \)

\[
K = CR
\]

**[Regular User Execution]:**

**Tuning the local model:**

1. \( w^k_{i} \leftarrow (w^k_{i}, D_{r(i)}) \)
2. \( B_k \leftarrow \text{(split} \ D_k \text{ into patches of size } B) \)
3. for \( i = 1, 2, \ldots \) do
4. for \( b \in B_k \) do
5. \( w^k_{i} \leftarrow w^k_{i} - \eta \nabla_{w^k_{i}}(w^k_{i}, b) \)

**Upload the local model each 24 h:**

6. for \( t = 0, 1, \ldots, T_C - 1 \) do:
7. \( \delta \leftarrow (K, \sigma) \)
8. if \( \delta > C \) then return \( w^i \)
9. else if
10. for each regular smart meter, do:
11. \( \Delta w_{i+1}, \xi \leftarrow (w^i) \)
12. \( S = \sum_{k} \xi_{k} \)
13. \( w^i_{i+1} \leftarrow w^i_{i} + \frac{1}{S} \left( \sum_{k} \Delta w^i_{i+1} / \max(1, \xi_{k}) + N(0, \sigma^2) \right) \)
14. end for
15. end if
16. end for
17. send \( w^i \)

**Obfuscating and send consumption data each 15min:**

18. Predict the state of the appliance:
19. \( Y^{\text{predict}} \leftarrow X \)
20. Obfuscation:
21. \( Y^{\text{RandomResponse}} \leftarrow Y \)
22. Recover:
23. \( X^{\text{Recover}} \leftarrow Y^{\text{RandomResponse}} \)
24. send \( Y^{\text{Recover}} \)

Therefore, in this case, the feature spaces and label spaces are not same

\[ X_{Ri} \neq X_{Rj}, Y_{Ri} \neq Y_{Rj} \]

where \( X \) denotes the feature space and \( Y \) denotes the label space.

Obviously, the user spaces are not same

\[ I_i \neq I_j, \forall D_i, i \neq j \]

**Learning procedure.** To the best of our knowledge, it is a complex task for existing HFL or vertical federated learning, because the user spaces, feature spaces, and label spaces are different. In order to solve the distribution difference, we do HFTL as follows.

First, the regular smart meter downloads the global model and other hyper-parameters such as feature space \( X_c \) and label space \( Y_c \) from electric power provider. Note that the sensitive parameters of user space \( I_R \) are not included. Second, the following simple objective function can be used to find a mapping for feature extraction

\[
\min_{\Phi} \left( \frac{(\text{DIST}(X^X, X^y; \Phi) + \lambda \Omega(\Phi))}{(\text{VAR}(X^X \cup X^y; \Phi))} \right)
\]

where \( \Phi \) is a mapping function, \( \text{DIST}() \) is a distribution difference metric, \( \Omega(\Phi) \) is a regularizer controlling the complexity of \( \Phi \), and \( \text{VAR} \) is the variance of instances.

Third, he runs one epoch of SGD training in his local dataset when using obfuscation mechanism. Last, sensitive smart meter \( i, i \in S \), computes \( \Delta w^f \)

\[
\Delta w^f = \nabla w^f = \sum_{i=1}^{j} w^f_i - w^f_j
\]

Note that performing the procedure with feature mapping is complex than regular users; however, for sensitive user with simple IoT devices, it is computationally acceptable. The Gaussian mechanism calibrated to the functions dataset sensitivity \( S_f \), therefore the Gaussian noise is defined as \( N(0, \sigma^2 S_f) \). Then, the regular users send \( f(d) + N(0, \sigma^2 S_f) \) to the electric provider.

Let \( w^0 \) denote the initial global model, then the objective function is

\[
\arg \min_{\Theta} L = \sum_{i=1}^{n} I(y_i, w_0(x_i))
\]

where \( I(a, b) \) denotes the loss function between \( a \) and \( b \), in this work we adopt it as cross-entropy loss; and \( \Theta \) denotes all the parameters to be learned, for example, the weight and bias in neural network, transition matrix and emission matrix in Markov model, or dictionary in sparse coding.

**Objective function.** The objective function of sensitive users is similar to that of regular users we have introduced in the previous section. To avoid repetition, the simple conclusion is given as follows.

In neural network, the objective function is

\[
\arg \min_{\Theta} L = \sum_{i=1}^{n} I(y'_i, w_i(x'_i))
\]

In sparse coding, the objective function is
Algorithm 3. Federated learning for sensitive smart grid or IoT users.
U is the number of sensitive users (smart meter) and \( \eta \) is the learning rate.

**Input:** pre-trained global model \( w^t \) from the grid, local cached data \( D_{(i)} \)

**[Sensitive User Execution]:**

**Tuning the local model:**
1. \( w^t_{(i)} \leftarrow \text{HIT} \left( w^t_{(i)}, D_{(i)} \right) \)
2. \( \min_0 \left( \text{DIST}(X^d_i, X^e_i), \Phi \right) + \lambda \Omega(\Phi) / (\text{VAR}(X^d_i \cup X^e_i, \Phi)) \)
3. \( B_i \leftarrow \text{(split } D_i \text{ into patches of size } 8) \)
4. for \( i = 1, 2, \ldots \) do
5. for \( b \in B_i \) do
6. \( w^t_b \leftarrow w^t_b - \eta \nabla F_b(w^t_b, b) \)

**Upload the local model each 24 h:**
7. send \( w^t_b \)

**Obfuscating and send consumption data each 15min:**
8. Predict the state of the appliance:
9. \( \gamma \) predict \( X \) _local model_
10. **Obfuscation:**
11. \( \gamma' \) RandomResponse \( \gamma \)
12. **Recover:**
13. \( X' \) Recover \( \gamma' \)
14. send \( \gamma' \)

\[
\arg \min_{\theta} L = \left\| x^d - \left[ \begin{array}{c} A_1 \\ \vdots \\ A_n \end{array} \right] \right\|
\]

Last, Algorithm 3 is given as follows:

**LDP analysis**

In this section, we adopt the DP method to analyze the privacy of IFed. The privacy of energy consumption data depends on specific obfuscation mechanism and not on the framework. We only discuss potential privacy disclosure risk due to users uploading their models applied to this framework.

First, we consider the privacy disclosure risk of sensitive users, which is different from conventional federated learning. Sensitive users do not upload their local model to anyone in this work, so that existing behavior inference works. Then, it can be considered that the sensitive users are completely secure. Next, in this section, we will discuss how risk exists on regular users.

Second, we explore the simplest case in which all regular users \( r_1, r_2, \ldots, r_l \) add the same standard normal noise to their model \( w^t_1, w^t_2, \ldots, w^t_l \) and generate new models \( w^t_1', w^t_2', \ldots, w^t_l' \)

**Theorem 1: Privacy loss for a given user.** The existing regular users \( r_1, r_2, \ldots, r_l \) add a given fixed noise \( \mathcal{N}(0, \sigma^2) \) to each local model, and then it is subjected to \( (\epsilon, \delta) \), where \( \delta' = \delta = \delta'' = \ldots \)

\[
\epsilon' = 0, \quad \delta' = \frac{4 e^{-\frac{\epsilon^2}{2}}}{5}
\]

This is the privacy loss for a given user, if the obfuscated model \( w'/r \) was snooped by adversary.

**Proof:** Let the sensitivity of \( \Delta f \) of \( w \) be \( \Delta f = \max \| f(w') - f(w') \| < 1 \), where \( w \) is adjacent to \( w' \), and then the Gaussian mechanism \( f(w') + N(0, \sigma^2) \) offers \((\epsilon, \delta)\)-DP,

\[
\delta \geq \frac{4 e^{-\frac{\epsilon^2}{2}}}{5}
\]

This is the privacy loss for a given user \( r_i \), if the obfuscated model \( w_i', w_i', \ldots, w_i' \) was snooped by adversary.

**Proof:** Let the sensitivity of \( \Delta f \) of \( w \) be \( \Delta f = \max \| f(w_i') - f(w_i') \| < 1 \), where \( w \) is adjacent to \( w' \). As \( f(w_i'), f(w_i'), \ldots, f(w_i') \) is \((\epsilon, \delta')\)-DP and here we choose averaging parameter for model aggregation, \( w^t = \sum_{i=1}^{n} w_i' \), and then the Gaussian mechanism \( f(w') + N(0, \sigma^2) \) offers \((\epsilon, \delta')\)-DP, where

\[
\delta \geq \frac{4 e^{-\frac{\epsilon^2}{2}}}{5}
\]

Now, we consider a more general approach, where each regular user adds the noise individually (see section “Key algorithms in IFed”).

**Theorem 3: Privacy loss for any user.** Electricity provider selects \( l \) regular users as active users from all regular users \( r_1, r_2, \ldots, r_l \), let \( q = l/t \). For any \( \epsilon < \frac{p_1}{q^2} \), our model aggregation is \((\epsilon, \delta')\)-DP for any \( \delta > 0 \) if we choose

\[
\delta \geq \frac{c_2}{e} \sqrt{\log \left( \frac{1}{\delta} \right)}
\]

**Proof:** It can be bounded as \( \alpha(\lambda) \leq \frac{q^2 \lambda^2}{\sigma^2} \). According to Abadi et al., \((\epsilon, \delta')\)-DP is

\[
\frac{q^2 \lambda^2}{\sigma^2} \leq \frac{\lambda e}{2} \exp \left( -\frac{\lambda e}{2} \right) \leq \delta
\]
Therefore, \( \lambda \leq \sigma^2 \log(1/q\sigma) \) when \( \varepsilon = c_1\sigma^2 \) and \( \delta = c_2(\sqrt{\log(1/\delta)})/\varepsilon \). The trade-off between data utility and privacy is mainly completed by advanced obfuscation mechanisms, and the detailed proves are seen\(^{24,26–28} \) in this article, we only discuss the privacy loss and privacy bound caused by iFed frameworks.

**Experiments**

In this section, we conduct experiments about iFed.

**Experimental setup**

**Dataset and baseline algorithm.** We use the reference energy disaggregation dataset (REDD)\(^{42} \) verification. It is a dataset containing the detailed power usage information which consists of whole home and circuit/device specific from six homes. Each home recorded the electricity consumption for a month and include the following: (1) the whole-home electricity signal at a high frequency (15 kHz) and (2) up to 24 individual circuits in the home, each labeled with its category of appliance or appliances, recorded at 0.5 Hz (plug-level monitors are recorded at 1 Hz).

We use SCRAPPOR\(^{20} \) and FHMM\(^{27} \) as baseline algorithm for iFed, which is already introduced in section “Preliminaries.” The parameters of SCRAPPOR are as follows: 300 signals as a batch that is the size of 15 min and 192 batches as a 300,192 matrix. Training iterations are 10,000. The parameters of FHMM are as follows: 300 signals as a batch that is the size of 15 min and max applicances as 10. In our experience, after 10,000 iterations, train takes about 50 min and the sparsity of dictionary achieves 99%.

**Metrics.** F1-score is a composition of four metrics employed in this article to evaluate iFed:

- **TP (true positive)**: positive in predication, positive in practice;
- **TN (true negative)**: positive in predication, negative in practice;
- **FP (false positive)**: negative in predication, positive in practice;
- **FN (false negative)**: negative in predication, negative in practice.

**Accuracy rate:** Precision = \( \frac{TP}{TP + FP} \)

**Recall rate:** Recall = \( \frac{TP}{TP + FN} \)

**F1-score:** F1-score = \( \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \)

**Experimental results**

We studied the effects of different algorithms and frameworks against NILM and compared the F1-score. Our experiments show that iFed provides a very good support for the advanced algorithms that achieve DP better in Power IoTs.

As shown in Table 1, we add noise to each of the aforementioned algorithms to generate the obfuscated data and then calculate the F1-scores of fridge, light, and washer dryer with Kim et al.,\(^{43} \) algorithm. Thus, by adding noise to the energy consumption signature, iFed has a better privacy-preserving level. In Figure 4, iFed shows the advantages of advanced obfuscation algorithms in DP in Power IoTs.

From an intuitive point of view, to achieve the same level of privacy protection, our framework with some advanced obfuscation mechanisms adds less noise than transitional schemes. Moreover, by adopting to our scheme, the computing burden of IoT terminal can be greatly reduced. In the following section, we will verify these two points by experiments.

| Algorithm               | Refrigerator | Microwave | Stove   | Washer |
|-------------------------|--------------|-----------|---------|--------|
| Raw data                | 0.75         | 0.82      | 0.73    | 0.57   |
| SCRAPPOR with iFed\(^{28} \) | 0.48         | 0.58      | 0.23    | 0.18   |
| FHMM algorithm with iFed\(^{28} \) | 0.49         | 0.58      | 0.23    | 0.19   |
| Barbosa’s algorithm     | 0.58         | 0.65      | 0.45    | 0.22   |
| Ankar’s algorithm       | 0.64         | 0.65      | 0.42    | 0.32   |

**Conclusion**

One fundamental challenge to LDP in Power IoTs is to show how to achieve the trade-off between utility and privacy while maintaining execution in naive IoT terminal. It is very different to the trade-off that has
brought a greater challenge. In this case, the key of design in this article is not the algorithm itself but a novel framework of LDP in Power IoTs. We presented a technique, a novel federated learning framework, to overcome this limitation. The technique uses a model that transmits substitute for high-risk data transmission, meanwhile the model transmission is satisfied with LDP. Our experiment demonstrated that this framework of federated learning is an alternative option for Power IoTs.

**Figure 4.** Obfuscation comparison between IFed and the existing algorithms: (a) original data collected from a smart meter without any obfuscation; (b) obfuscation data using Barbosa’s algorithm, (c) obfuscation data using SCRAPPOR, and (d) obfuscation data using SCRAPPOR through IFed.

**Figure 5.** Utility comparison. This figure shows that the data loss of advanced obfuscation mechanisms with IFed is lower than other regular obfuscation mechanisms. The solid line represents the error of advanced obfuscation mechanism with IFed and the dotted line represents the error of Barbosa’s scheme.
Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Major Technical Innovation Project of Hubei (Grant No. 2018AAA046), National Natural Science Foundation of China (Grant No. 61872431), and Applied Basic Research Project of Wuhan (Grant No. 2017060201010162).

ORCID iDs

Hui Cao https://orcid.org/0000-0001-5136-9084
Xingxing Xiong https://orcid.org/0000-0002-6347-7805

References

1. Al-Garadi M, Mohamed A, Al-Ali A, et al. A survey of machine and deep learning methods for Internet of Things (IoT) security. 2018, https://arxiv.org/abs/1807.11023
2. Qiu J, Tian Z, Du C, et al. A survey on access control in the age of internet of things. IEEE Int Thing J. Epub ahead of print 24 January 2020. DOI: 10.1109/JIOT.2020.2969326.
3. Saleem Y, Crespi N, Rehmani MH, et al. Internet of Things-aided smart grid: technologies, architectures, applications, prototypes, and future research directions. IEEE Access 2019; 7: 62962–63003.
4. Du X, Guizani M, Xiao Y, et al. Defending DoS attacks on broadcast authentication in wireless sensor networks. In: Proceedings of the IEEE international conference on communications (ICC 2008), Beijing, China, 19–23 May 2008.
5. Marmol F, Sorge C, Ugus O, et al. Do not snoop my habits: preserving privacy in the smart grid. IEEE Commun Mag 2012; 50(5): 166–172.
6. Hart G. Nonintrusive appliance load monitoring. Proc IEEE 1992; 18(12): 1870–1891
7. Guan Z, Zhang Y, Zhu L, et al. EFFECT: an efficient flexible privacy-preserving data aggregation scheme with authentication in smart grid. Sci Chin Inf Sci 2019, 62(3): 032103.
8. Tian Z, Shi W, Wang Y, et al. Real time lateral movement detection based on evidence reasoning network for edge computing environment. IEEE Trans Ind Inf 2019; 15(7): 4285–4294.
9. Tian Z, Luo C, Qiu J, et al. A distributed deep learning system for web attack detection on edge devices. IEEE Trans Ind Inf 2020; 16(3): 1963–1971.
10. Klemenjak C and Goldsborough P. Non-intrusive load monitoring: a review and outlook. In: Proceedings of the IEEE international conference on consumer electronics (ICCE), Las Vegas, NV, 7–11 January 2016, pp.2199–2210.
11. Sun M and Tay W. On the relationship between inference and data privacy in decentralized IoT networks. Trans Inf Foren Sec 2020; 15: 852–866.
12. Andres ME, Bordenabe NE, Chatzikokolakis K, et al. Geo-indistinguishability: differential privacy for location-
based systems. In: Proceedings of the ACM SIGSAC conference on computer communications security, Berlin, 4–8 November 2013, pp.901–914.

13. Dwork C and Roth A. The algorithmic foundations of differential privacy. Found Trend Comput Sci 2014; 9(3–4): 211–407.

14. Dwork C, McSherry F, Nissim K, et al. Calibrating noise to sensitivity in private data analysis. In: Proceedings of the theory of cryptography: third theory of cryptography conference (TCC 2006). New York, NY, 4–7 March 2006, pp.265–284.

15. Guan Z, Liu X, Wu L, et al. Cross-lingual multi-keyword rank search with semantic extension over encrypted data. Inf Sci 2020; 514: 523–540.

16. Li F, Luo B and Liu P. Secure information aggregation for smart grids using homoeorototic encryption. In: First IEEE international conference on smart grid communications (Smart Grid Comm), Gaithersburg, 4–6 October 2010, pp.327–332.

17. Bohli J-M, Sorge C and Ugus O. A privacy model for smart metering. In: IEEE international conference on communications workshops (ICC), Cape Town, South Africa, 23–27 May 2010, pp.1–5.

18. Li M, Sun Y, Lu H, et al. Deep reinforcement learning for partially observable data poisoning attack in crowd sensing systems. IEEE Int Thing J 2020. DOI: 10.1109/JIOT.2019.2962914.

19. Tan O, Gunduz D and Poor HV. Increasing smart meter privacy through energy harvesting and storage devices. IEEE J Select Areas Commun 2013; 31(7): 1331–1341.

20. Choi W, Tomei M, Rodrigo J, et al. Guaranteeing local differential privacy on ultra-low-power systems. In: 2016 ACM/IEEE 43rd annual international symposium on computer architecture (ISCA), Seoul, South Korea, 18–22 June 2016, pp.14–26.

21. Kandukuri BR, Paturi VR and Rakshit A. Cloud security issues. In: Proceedings of the 2009 IEEE international conference on services computing, Bangalore, India, 21–25 September 2009, pp.517–520. Washington, DC: IEEE Computer Society.

22. Finster S and Baumgart I (eds). Privacy-aware smart metering: a survey. In: IEEE communications surveys & tutorials (volume 17, issue 2, Second quarter). New York: IEEE, 2015, pp.1088–1101.

23. Erlingsson U, Pihur V and Korolova A. RAPPOR: randomized aggregatable privacy-preserving ordinal response. In: Proceedings of the ACM SIGSAC conference on computer communications security, Scottsdale, AZ, 3–7 November 2014, pp.1054–1067.

24. Kairouz P, Bonawitz K and Ramage D. Discrete distribution estimation under local privacy. In: Proceedings of the 33rd international conference on machine learning, New York, NY, 19–24 June 2016, pp.2436–2444.

25. Murakami T and Kawamoto Y. Utility optimized local differential privacy mechanisms for distribution estimation. USENIX Sec 2019; 19: 1877–1894.

26. Guan Z, Lv Z, Du X, et al. Achieving data utility-privacy tradeoff in internet of medical things: a machine learning approach. Futur Gen Comput Syst 2019; 98: 60–68.

27. Cao H, Liu S, Wu L, et al. Achieving differential privacy against -intrusive load monitoring in smart grid: a fog computing approach. Concur Comput Pract Exp 2018; 4: 1–14.

28. Cao H, Liu S, Wu L, et al. SCRAPPER: an efficient privacy-preserving algorithm base on sparse coding for information-centric IoT. IEEE Access 2018; 6: 663143–663154.

29. Yang Q, Liu Y, Chen T, et al. Federated machine learning: concept and applications. ACM Trans Intell Syst Tech 2019; 10(12): 1–19.

30. Samarakoon S, Bennis M, Saad W, et al. Distributed federated learning for ultra-reliable low-latency vehicular communications. IEEE Trans Commun. arXiv:1807.08127.

31. Faustine A, Mvungi N, Kaigajez S, et al. A survey on non-intrusive load monitoring methods and techniques for energy disaggregation problem, https://arxiv.org/abs/1703.00785.

32. Zoha A, Gluhak A, Imran M, et al. Non-intrusive load monitoring approaches for disaggregated energy sensing: a survey. Sensors 2012; 12: 16838–16866.

33. Duchi JC, Jordan MI and Wainwright MJ. Local privacy and statistical minimax rates. In: Proceedings of the IEEE symposium on foundations of computer science, Berkeley, CA, 26–29 October 2013, pp.429–438.

34. Xu C, Ren J, Zhang D, et al. Distilling at the edge: a local differential privacy obfuscation framework for IoT data analytics. IEEE Commun Mag 2018; 56(8): 20–25.

35. Kawamoto Y and Murakami T. Local obfuscation mechanisms for hiding probability distributions. In: Sako K, Schneider S and Ryan P (eds) Computer Security—ESORICS 2019 (Lecture Notes in Computer Science, vol.11735). Cham: Springer, 2019, pp.128–148.

36. Bonawitz K, Ivanov V, Kreuter B, et al. Practical secure aggregation for privacy-preserving machine learning. In: Proceedings of the 2017 ACM SIGSAC conference on computer and communications security, Dallas, TX, 30 October–3 November 2017, pp.1175–1191.

37. Du X, Xiao Y, Ci S, et al. A routing-driven key management scheme for heterogeneous sensor networks. In: Proceedings of the IEEE international conference on communications, Glasgow, 24–28 June 2007.

38. Xu J and Wang F. Federated learning for healthcare informatics. ACM Trans Comput Healthc. arXiv:1911.06270v1.

39. Brendan MH, Ramage D, Talwar K, et al. Learning differentially private recurrent language models. In: International conference on learning representations, Vancouver, BC, Canada, 30 April–3 May 2018.

40. Pan SJ and Yang Q. A survey on transfer learning. IEEE Trans Knowl Data Eng 2020; 22(10): 1345–1359.

41. Abadi M, Chu A, Goodfellow I, et al. Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, Vienna, 24–28 October 2016, pp.308–318.

42. Koller JZ and Johnson MJ. REDD: a public data set for energy disaggregation research. In: Proceedings of the workshop on data mining applications in sustainability, San Diego, CA, 21 August 2011, pp.59–62.

43. Kim H, Marwha M, Arlitt MF, et al. Unsupervised disaggregation of low frequency power measurements. In Proceedings of the 11th SIAM international conference on data mining, 2011, pp.747–758. Society for Industrial and Applied Mathematics.