Differential Privacy for Deep Learning-based Online Energy Disaggregation System

Xiao-Yu Zhang, Stefanie Kuenzel

Department of Electronic Engineering
Royal Holloway, University of London
Egham Hill, Egham TW20 0EX, UK

{Xiaoyu.Zhang & Stefanie.Kuenzel}@rhul.ac.uk

***Abstract*— Online energy disaggregation is an advanced technology that can help both consumers and the utility to implement load components analysis, enhancing the reliability of demand-side management. However, the online system requires continuous access to personal electricity data to train an online machine learning model, which would infer personal information. In this paper, we introduce a privacy-preserving online energy disaggregation system, the online model and training data can be protected by adding Gaussian noise to the model during the training process.**

Keywords*— *deep learning, smart meter, cloud computing, energy disaggregation, differential privacy

I. Introduction

The applications of advanced machine learning algorithms and big data technology in the power system enhance the efficiency and reliability of demand-side management (DSM). Deep neural network (DNN) based technologies, such as energy disaggregation and load forecasting, have attracted researchers' interest for the high performance in peak load shaving and valley filling [1], [2], [3], [4]. Moreover, with intelligent real-time sensing devices such as the smart meter, and the smart plug installed worldwide [8], it is possible for the utility to collect individuals' power consumption data (active/ reactive power, current, voltage, and harmonics) with high sampling frequency. However, to train the online energy disaggregation system, the operator would collect a large amount of personal smart meter data from individuals. The smart meter data is highly sensitive personal data which would leak personal behaviours and activities inside a house (how many residents stay in the house, when people leave home, what the residents are doing at particular times, such as sleeping, bathing, watching TV, washing clothes, etc.). It is essential to present a privacy-preserving energy disaggregation system to protect the sensitive data and meanwhile provide services to consumers.

The online energy disaggregation, or online Nonintrusive load monitoring (NILM), is a technique that disaggregates overall household power consumption into the consumption of individual appliances, providing detailed power usage information to end-users. The state-of-the-art NILM technique relies on deep learning algorithms (Long Short-Term Memory Recurrent Neural Network, Convolutional Neural Network, et al.), providing high accuracy results to consumers. While many people believe the training process of the deep learning algorithm is a “black box” and the parameters inside the neural network cannot be inferred by others, the fact is an adversary can easily obtain the model parameters and simulate the training process through privacy inference attacks such as membership inference [5] and model inversion attack [6]. As a consequence, the adversary can further extract the details of the training data.

Differential privacy is a mathematical definition of privacy, an algorithm can be treated as differential private if an adversary cannot distinguish whether an individual's data is included/not included in a dataset by observing the output of the algorithm [7]. Recent works have proven differential privacy can be combined with a neural network to provide privacy-preserving services such as cloud services provided by Google and Amazon [8], [9].

In this paper, we first introduce an online energy disaggregation system, the system can provide efficient services to both demand-side managers and consumers. We also analyse the privacy risks of the system and develop an adversary that can infer personal information from the system. We then propose a privacy-preserving deep learning algorithm that satisfies (ε, δ) differential privacy, while the privacy level ε quantifies the privacy leakage and can be adjusted to refer to the privacy requirement of the utility.

II. Related Works

A. Energy Disaggregation

At the earliest stage of a DSM process, the demand-side manager needs to design an energy reduction/ increase plan of the targeted area. The plan requires the knowledge of the portions of load categories of each consumer at a given timestamp via energy disaggregation. A technology called nonintrusive load monitoring (NILM) is adopted by the companies to implement the energy disaggregation [4], [10]. The smart meter would collect household-level data and upload the data to the online NILM model via a wireless communication network, then the model would evaluate the percentages of different appliances and share with the demand-side manager. Deep learning-based NILM would adopt a DNN model to implement the disaggregation, both long-short term memory (LSTM) and convolutional neural network (CNN) algorithm already reach a higher accuracy than conventional methods (such as Hidden Markov Model) [10].

B. Differential Privacy

Proposed by C. Dwork in 2006, differential privacy is a technology to protect an individual's identification information by adding random noise over the original aggregated data, every individual has little effect on the final result [7], [11], [12]. In this case, the adversary cannot distinguish the change of the aggregated data with/without one individual data. There are serval noise addition mechanisms available in the literature [13], including Laplace mechanism, Exponential mechanism, and Gaussian mechanism. The privacy level, ε , is guaranteed via the above noise addition mechanism, and lower ε , the higher the privacy level can be achieved.

Definition 1. $\mathfrak{R}$ is a random function that transforms input β to a random output $\mathfrak{R}(\beta)$.

Definition 2. $d(\beta,\beta')$, which is the distance between two neighboring datasets, represents the minimum number of individual samples required to shift dataset β to β'.

Definition 3. The global sensitivity, S_f, is the maximum difference between the outputs of two neighboring datasets β and β'. S_f also determines the overall noise to be added into the DP mechanism.

$$\Delta f = \max_{d(\beta,\beta')=1} \|f(\beta) - f(\beta')\| \tag{1}$$

Definition 4. The Gaussian privacy mechanism denoted $\mathfrak{R}$, is defined as f plus noise term $\mathcal{N}$. While $\mathcal{N}$ is the Gaussian distribution with mean 0 and standard deviation ${S_f}^2\sigma^2$.

$$\mathfrak{R}(\beta) \triangleq f(\beta) + \mathcal{N}(0, \Delta f^2\sigma^2) \tag{2}$$

Definition 5. A randomized function $\mathfrak{R}$ satisfies (ε, δ) privacy $\mathbb{P}_{\mathbb{R}}$ for any two neighboring datasets β and β':

$$\mathbb{P}_{\mathbb{R}}[\mathfrak{R}(\beta) \in \xi] \le e^{\varepsilon}\mathbb{P}_{\mathbb{R}}[\mathfrak{R}(\beta') \in \xi] + \delta \tag{3}$$

where ξ denotes all possible outcomes in range $\mathfrak{R}$, and δ is the possibility that the differential privacy is broken, in this paper, we select 10^{-5} as δ.

C. Deep Neural Network

A DNN is an artificial neural network with *N* multilayers, the larger the number of layers, the deeper computation the network can achieve. Given a training corpus $(\boldsymbol{X},\boldsymbol{Y})$, the output of the network a_N is:

$$a_N(x;\ \theta_{1,\cdots,N}) = f_N(f_{N-1}(\cdots f_1(x,\theta_1),\theta_{N-1}),\theta_N) \tag{4}$$

where x is the input of the model, θ_n are the parameters of the nth layer, $f_n(x,\theta)$ is the linear/ nonlinear function of the neuron (sigmoid, tanh, SoftMax, et. al).

A loss function $\mathcal{L}$ is adopted to calculate the mismatch between ground truth y and a_N. The purpose of the DNN is to find the optimal parameters of the model θ^*that minimize the $\mathcal{L}$ throughout the whole training process:

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{(x,y)\in(X,Y)} \mathcal{L}(y, a_N(x;\theta_{1,\cdots,N})) \tag{5}$$

$$\theta^* \leftarrow argmin_{\theta}\mathcal{L}(\theta) \tag{6}$$

Since DNN has a large amount of training data and the topological structure is extremely complex, it is difficult to find the optimal solution. The stochastic gradient descent (SGD) is introduced to solve the problem. Instead of selecting all training data to train, SGD only randomly picks up a group of training data, which is a term called 'batch' B , to do iteration. The gradient of the batch g_B is used to estimate the original gradient $\nabla_{\theta}\mathcal{L}(\theta)$, see equation (7):

$$g_B = \frac{1}{B}\sum_{x\in B} \nabla_{\theta}\mathcal{L}(\theta, x) \tag{7}$$

D. Deep Learning with Differential Privacy

To counter the above attacks, differential privacy deep learning is proposed, it combines deep neural networks with differential privacy [8], [9] . Instead of masking the final result evaluated by the DNN model, differential private stochastic gradient descent (DP-SGD) mechanism adds noise during the whole training process, protecting all parameters as well as training data inside the model. The mechanism also introduces a sophisticated approach that can adjust the level of privacy ε smoothly refer to the privacy requirment.

III. System Overview

In this section, we first discuss the current online energy disaggregation system and the vulnerability of the system. Based on the system, we design an adversary model that can infer private information from the system.

A. Online Energy Disaggregation Model

As an important part of the intelligent DSM system, the online energy disaggregation system consists of four parts steps (shown in Fig. 1), which are described as follows in detail:

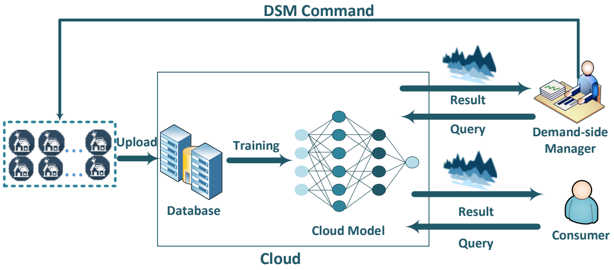


Fig. 1. The Architecture of Online Energy Disaggregation System.

1)Data Collection: Since the load components, climate, and electricity behaviours vary a lot in different areas, it is essential training the DNN model with local residential data rather than a public dataset. The installation of the smart meter enables real-time bidirectional communication between the utility and the consumers. Hence, the online database used to train the DNN is built by continuously collecting power consumption data from local smart meters.

2)Model Training: The online NILM model would utilize either the supervised learning algorithm [10] (both household-level data and appliance-level data is required for training, smart plugs should be installed randomly in sample houses to construct the appliance dataset, see [14]) or unsupervised learning algorithm [15] (only household-level data is required) to train the model. The input of the model would be the household-level data (active/reactive power, voltage, current, and potentially harmonics) and the output of the model would be the disaggregated power of individual appliances.

3)Query Process: The clients of the online service are either demand-side managers or the consumer. The services the model can provide include four categories: (1) Event detection; (2) Feature extraction; (3) Clustering; (4) Matching [15]. The consumer can send a query to obtain a detailed report about the energy consumption of appliances inside his house via the smart meter. The demand-side managers also need to have permission to access the online system, the managers can design an energy increase/ reduction plan referring to the load components information evaluated by the system.

4)DSM: With the energy plan designed in the last step, the demand-side manager can send DSM command to the local residences, this process is also called load control. Both direct load control and indirect load control are discussed in previous literature [16].

B. Adversary Model and the Definition of 'Privacy'

While the above online system can provide efficient services to the consumers and the utility, there is vulnerability related to privacy. The adversary in this paper is strong enough to adopt inference attacks (membership inference attacks or model inversion attacks) to obtain all model parameters, and further infer high-sensitive information from the training data [17], [18]. We define 'privacy' as the information that can help the adversary identify the individual. In the power system, power usage data is highly related to human behaviour. By constantly inferring the data, the adversary could have an overview of the consumer's behaviour inside the house.

An adversary inference process is shown in flowchart Fig.2, the cloud DNN model is trained with individuals' power consumption data measured by smart meters installed at their home.

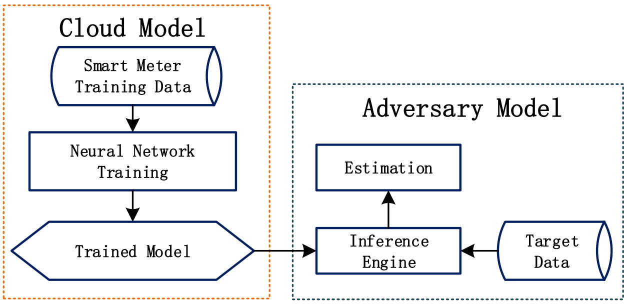


Fig. 2. Flowchart of adversary inference process.

IV. Privacy-Preserving Online Energy Disaggregation System

Recall the original system shown in Fig. 1, the proposed privacy-preserving energy disaggregation system adds Gaussian noise to the second step — Model Training. DP-SGD is applied to provide DP protection to the online energy disaggregation model.

A. Differential Private Stochastic Gradient Descent Algorithm

We adopt the DP-SGD algorithm proposed by M. Abadi [8]. Considering a DNN with N training data, in each noise adding process, a group of *L* data is selected with a selection possibility *q (q=L/N)*, and a batch size *B* from *L* is selected for every iteration. For *i* ∈ T (T is the number of training steps):

1) Compute Gradient: The gradient of tth training steps $g_t(x_i)$ is computed via Equation (7);

2) Clipping Gradient: The purpose to clip gradients is to reduce the sensitivity of each training data to the model optimizer. A threshold C is introduced to limit the maximum Euclidean norm of the training data. The clipped gradient $\bar{g}(x_i)$ is:

$$\bar{g}(x_i) = \frac{g(x_i)}{\max(1, \|g(x_i)\|_2/C)} \tag{8}$$

3) Adding Random Noise to the Gradient: A random Gaussian is added to the clipped gradient, each step is (ε, δ)-privacy:

$$\tilde{g}(x_i) = \frac{1}{L}\left(\sum_{x\in L}\bar{g}(x_i) + \mathcal{N}(0, \Delta f^2\sigma^2)\right) \tag{9}$$

Where $\sigma = \frac{\sqrt{2log1/\delta}}{\varepsilon}$.

4) Update Parameters after each training step t:

$$\theta_{t+1} = \theta_t - \alpha\tilde{g}_t(x_i) \tag{10}$$

B. Structure of the Model

We adopted a supervised DNN model as our online model. 60000 training data and 20000 testing data are fed to the DNN model. Detailed structure is shown in Table I. In the model, there are four fully connected layers with dropout layers. The function of the dropout layer is to avoid overfitting. As for the output layer, a SoftMax activation function is used to classify the state of the appliance.

TABLE I
The Structure of the Online DNN Model

Layer	**Activation Function**	**Output Shape**	**Parameters**
Fully connected	RELU	(1, 256)	512
Dropout	—	(1,256)	0
Fully connected	RELU	(1,512)	131584
Dropout	—	(1,512)	0
Fully connected	RELU	(1,1024)	525312
Dropout	—	(1,1024)	0
Fully connected	SoftMax	(1,3)	3075

C. Model Hyperparameters

The hyperparameters in the model include norm clipping, noise level, learning rate. In this paper, we utilize defeat settings in the TensorFlow Privacy library [19]. The learning rate α is chosen as 0.25, the gradient norm boundary C is chosen as 1.5, the lot size L is chosen as 250, and we selected the batch size B as 50. There is already a rich literature illustrate the model hyperparameters selection [8] [20] [19], so we skip the detailed demonstration process.

D. Benchmark and Baseline

Although the privacy level is adjustable depending on the utility requirement, there is no clear boundary between good and bad privacy performance. To present the simulation result clearly, we use the definition of "privacy levels" in [20], which is shown in Table II.

The baseline of the energy disaggregation system performance is shown in Table III, which is evaluated by J. Kelly and W. Knottenbelt [10]. They compute the accuracy rates of the model without DP.

TABLE II
Benchmark of Differential Privacy Level [20]

Privacy Level	**ε**
Poor	≥ 3
Fine	1~3
Good	≤ 1

TABLE III
THE ACCURACY OF DNN-BASED ENERGY DISAGGREGATION MODEL WITHOUT DIFFERENTIAL PRIVACY [10]

Appliance	**Accuracy(%)**
Microwave	98
Fridge	81
Dish Washer	70
Average	83

V. IMPLEMENTATION

A. Dataset

In this paper, we use "The Reference Energy Disaggregation Data Set (REDD)" [21]. REDD contains 6 houses and 20 appliances with an interval resolution of 3s. REDD is a desirable dataset for simulation for its high sampling frequency and low noise rate.

B. Hardware and Software

The simulation and computation are implemented on a Dell laptop equipped with Core i7-7700HQ CPU and 8GB RAM. The deep learning algorithm was run on Python 3.6, the Tensorflow framework is adopted to train the DNN model, and the Tensorflow Privacy library [19] is adopted to enable implementation of DP on DNN.

C. Dataset Pre-processing

The original appliance data is the power consumption of appliances in kW, we transfer the power into the state information. There are two advantages to implement the transformation: (1) this method reduces the computation volume of the cloud computer; (2) the transformed curve is more flatten compared to the original power consumption curve, see Fig. 3. It can hide more detailed characteristic information which would reveal private information. As shown in Table III, although most appliances only have two states, which are "ON" and "OFF", an appliance may have multi states, such as the dishwasher. So, the function of the DNN model turns to a classification problem, to identify the current state of the target appliance. After obtaining the state database, we further transfer the data to one-hot encoded data, which is widely adopted in classification tasks.

TABLE IV
THE POWER AND STATES OF APPLIANCES [10] [15]

Appliance	**States**	**Power Each State (Watts)**
Microwave	2	S1: 0 S2: 500 S3: 1500
Fridge	2	S1: 0 S2:100
Dishwasher	3	S1: 0 S2:100 S3:1500

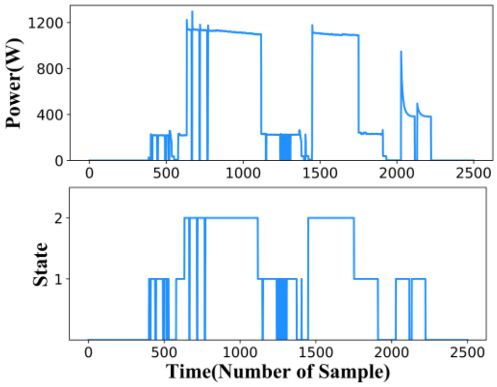


Fig. 3. The curve of original (upper) and transformed (lower) appliance-level data (Dishwasher as example).

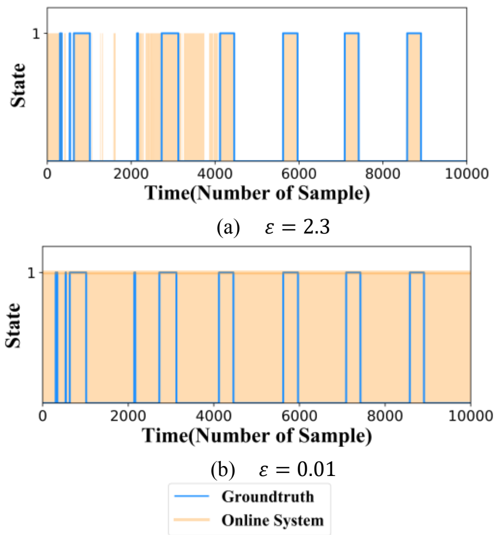
VI. SIMULATION AND RESULT

In this section, we implement case studies to verify the capability of the privacy-preserving energy disaggregation system. As described in Table IV, three appliances, microwave oven, dishwasher, and fridge are adopted in this section. We adjust the privacy level ε gradually and observe how ε influences the performance of the online system. Figure 4 shows the performance of the online DNN model with different ε. The blue curve represents the ground truth, while the orange shallow represents the result evaluated by the model. It is observed that with the decrease of ε, the performance gets worse.

The privacy-accuracy curve is shown in Fig. 5, while the online system with differential privacy can reach an average accuracy of 85%, the system with differential privacy has worse performance. Although the larger noise added to the DNN contributes to a better privacy performance, it sacrifices accuracy at the same time. Hence, a good privacy level should have a trade-off between utility and privacy. From the figure, a ε between 1 and 3 is desirable.

VII. CONCLUSION AND DISCUSSION

In this paper, we develop a privacy-preserving online energy disaggregation system to deal with the privacy issues raised by cloud computing and online DNN. We first introduce the online energy disaggregation system and highlight the functionalities it brings to both the utility and consumers, especially in DSM. A DP-SGD algorithm is applied to the training process of the online model, we add random Gaussian noise to the gradient of every training step, satisfy (ε, δ)-differential privacy. We also implement case studies via public power consumption dataset REDD, from the result of the simulation, it is found that with the privacy level ε reduces (noise level increases), the accuracy rate of the online system decreased gradually. Once the privacy level is quantified, the utility can provide multi-level privacy protection refers to consumers' requirements.



(a) $\varepsilon = 2.3$

(b) $\varepsilon = 0.01$

Fig. 4. Performance of the online system with different ε (Fridge as example).

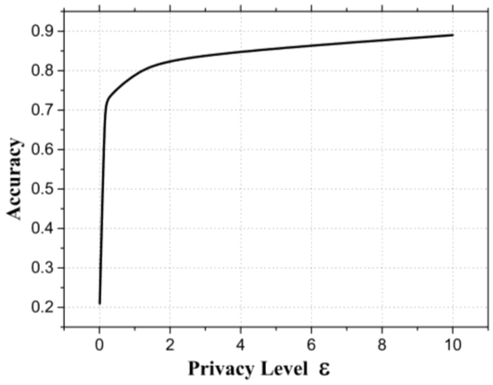


Fig. 5. Privacy-Accuracy Curve (Average value of appliances).

REFERENCE

[1] H. Shi, M. Xu and R. Li, “Deep Learning for Household Load Forecasting—A Novel Pooling Deep RNN,” IEEE Transactions on Smart Grid, vol. 9, no. 5, pp. 5271-5280, 2018.

[2] M. Rafiei, T. Niknam, J. Aghaei, M. Shafie-Khah and J. P. S. Catalão, “Probabilistic Load Forecasting Using an Improved Wavelet Neural Network Trained by Generalized Extreme Learning Machine,” IEEE Transactions on Smart Grid, vol. 9, no. 6, pp. 6961-6971, 2018.

[3] V. Singhal, J. Maggu and A. Majumdar, “Simultaneous Detection of Multiple Appliances From Smart-Meter Measurements via Multi-Label Consistent Deep Dictionary Learning and Deep Transform Learning,” IEEE Transactions on Smart Grid, vol. 10, no. 3, pp. 2969-2978, 2019.

[4] J. M. Gillis, S. M. Alshareef and W. G. Morsi, “Nonintrusive Load Monitoring Using Wavelet Design and Machine Learning,” IEEE Transactions on Smart Grid, vol. 7, no. 1, pp. 320-328, Jan. 2016.

[5] R. Shokri, M. Stronati, C. Song and V. Shmatikov, “Membership Inference Attacks Against Machine Learning Models,” in 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, 2017.

[6] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, and D Song, “The Secret Revealer: Generative Model-Inversion Attacks Against Deep Neural Networks,” arXiv, 17 Nov 2019.

[7] P. Barbosa, A. Brito, and H. Almeida, “A Technique to provide differential privacy for appliance usage in smart metering,” Information Sciences, pp. 355-367, 20 November 2016.

[8] Abadi, Martin et al., "Deep Learning with Differential Privacy," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS’16, 2016.

[9] M. A. P. Chamikara and P. Bertok and I. Khalil and D. Liu and S. Camtepe, "Local Differential Privacy for Deep Learning," arXiv, 2019.

[10] J. Kelly and W. Knottenbelt, “Neural NILM: Deep Neural Networks Applied to Energy Disaggregation,” in ACM BuildSys'15, Seoul, 2015.

[11] C. Dwork, “Differential Privacy,” in Proceedings of the ICALP'06 Proceedings of the 33rd international conference on Automata, Languages and Programming - Volume Part II, Venice, Italy, 2006.

[12] Z. Ji, Z. C. Lipton, and C. Elkan, "Differential Privacy and Machine Learning:a Survey and Review," 2014.

[13] M. Hassan, M. Rehmani, and J. Chen, “Differential Privacy Techniques for Cyber Physical Systems: A Survey,” 2019.

[14] “Dataport,” Pecan Strret, [Online]. Available: https://dataport.pecanstreet.org/. [Accessed 29 October 2019].

[15] M. Mengistu et. al, “A Cloud-Based On-Line Disaggregation Algorithm,” IEEE TRANSACTIONS ON SMART GRID, pp. 3430-3439, May 2019.

[16] Karwe M., Strüker J., “A Survey on Privacy in Residential Demand Side Management Applications,” in Smart Grid Security, Springer, 2014.

[17] Fredrikson, Matt, Somesh Jha, and Thomas Ristenpart, “Model inversion attacks that exploit confidence information and basic countermeasures,” in Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 2015.

[18] R. Shokri, M. Stronati, C. Song and V. Shmatikov, “Membership Inference Attacks Against Machine Learning Models,” in 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, 2017.

[19] Google LLC, "TensorFlow Privacy," 2019. [Online]. Available: https://github.com/tensorflow/privacy. [Accessed 7 November 2019].

[20] A. Liu et. al, “Differential privacy for eye-tracking data,” in ETRA '19 Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications, Denver, Colorado, 2019.

[21] J. Z. Kolter and M. J. Johnson, "REDD: A Public Data Set for Energy Disaggregation Research," Artificial Intelligence, no. 25, 2011.