Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 19(4): 3350–3368.
DOI: 10.3934/mbe.2022155
Received: 08 October 2021
Revised: 21 December 2021
Accepted: 09 January 2022
Published: 24 January 2022

Research article

Mitigating consumer privacy breach in smart grid using obfuscation-based generative adversarial network

Sanket Desai*, Nasser R Sabar, Rabei Alhadad, Abdun Mahmood and Naveen Chilamkurti

Department of Computer Science & I.T., La Trobe University, Melbourne, VIC 3083, Australia

* **Correspondence:** Email: s6desai@students.latrobe.edu.au.

Abstract: Smart meters allow real-time monitoring and collection of power consumption data of a consumer's premise. With the worldwide integration of smart meters, there has been a substantial rise in concerns regarding threats to consumer privacy. The exposed fine-grained power consumption data results in behaviour leakage by revealing the end-user's home appliance usage information. Previously, researchers have proposed approaches to alter data using perturbation, aggregation or hide identifiers using anonymization. Unfortunately, these techniques suffer from various limitations. In this paper, we propose a privacy preserving architecture for fine-grained power data in a smart grid. The proposed architecture uses generative adversarial network (GAN) and an obfuscator to generate a synthetic timeseries. The proposed architecture enables to replace the existing appliance signature with appliances that are not active during that period while ensuring minimum energy difference between the ground truth and the synthetic timeseries. We use real-world dataset containing power consumption readings for our experiment and use non-intrusive load monitoring (NILM) algorithms to show that our approach is more effective in preserving the privacy level of a consumer's power consumption data.

Keywords: smart grid; privacy preserving; generative adversarial network; consumer profiling

1. Introduction

Over the past decade, advances in the industrial and social sectors have drastically increased the demand for energy consumption. For example, Energy International Agency (EIA) projects a nearly 50% increase in world energy usage by 2050, led by growth in Asia from 2018 to 2050. The buildings sector, which includes residential and commercial infrastructure, is estimated to increase by 65% in energy consumption between 2018 and 2050 i.e., from 91 quadrillions to 139 quadrillions British Thermal Unit (BTU) [1]. The global CO_2 emission will increase more than double by 2050 while the global investment in electrical grid infrastructure is estimated to be around \$6 trillion by 2030 [2]. To meet this ever-increasing demand for energy supply, the need for efficient use of energy resources,

reduced carbon emission, and integration of multiple sources of renewable energy, required a new electrical grid to incorporate the digital and computing technologies to automate and manage the energy supply needs of the 21*st* century.

A smart grid is an electrical network that integrates information and communication technologies for efficient distribution and consumption of energy resources. The integration of various communication and data processing capabilities transforms the traditional electrical grid into a revolutionized power system, enabling information flow between different entities such as metering, substations, distributions, transmission, and generation [3]. With this increased availability of communication and computing resources, the smart grid has enhanced benefits and potential unknown to the traditional electrical network. For instance, a smart grid with its broad range of grid-side and consumer-side applications, enables monitoring of energy consumption data, demand response, dynamic pricing, and different information messages via its smart infrastructure. It also enables collecting and processing various types of energy related data through its smart infrastructure consisting of different entities such as grid sensors, wide-area monitoring, distribution energy management systems, etc. [4].

One such important asset of the smart grid is known as Advanced Metering Infrastructure (AMI). The AMI is made up of a set of smart meters, communication modules, local area network (LAN), data concentrator (DC), wide area network (WAN), software, and hardware of central system [5]. The AMI allows two-way communication between the consumer's smart meter and the energy supplier for measuring periodic or on-demand fine-grained energy consumption data. This fine-grained energy consumption data as feedback helps reduce cost and reduce consumption by up to 20% through efficient energy management. The European Parliament and Council of the European Union has taken one such initiative under the EU Directive 2006/32/EC to provide accurate measuring and actual "time of use" of energy consumption to the energy consumers [6].

While such detailed energy consumption feedback benefits economically and ecologically for involved stakeholders, smart meters allow massive energy information flow between consumers and suppliers, causing a potential threat to consumers' privacy. This sensitive energy information in collaboration with algorithms such as NILM [7], can help third parties deduce a consumer's daily routine, appliance usage, working hours, meal hours, occupants present on premises or any medical equipment in usage and even living habits such as the time when TV is watched.

In 2009, the Federal Bureau of Investigation's Cyber Intelligence investigated a widespread incident of power theft related to smart meters. It was found that the miscreants hacked in to the smart meters and reprogrammed the power consumption settings, resulting in a loss of $US 400 million annually for the Puerto Rico utility [8]. Furthermore, in 2007, the Austin Energy/Austin Police conducted a warrantless surveillance program where consumer usage information was provided to find marijuana growing operations. Furthermore, law enforcement agencies might use the data as real-time surveillance [9]. Figure 1 shows how NILM enables the identification of individual appliances using various machine learning algorithms from a single aggregate power consumption reading of a consumer's premise. Various entities such as law enforcement agencies, marketing agencies, and malicious users may misuse this fine-grained data to profile a consumer and jeopardize their privacy or to achieve unfair business strategies.

Recent research findings [3] on various privacy-preserving schemes and their implementation conclude that there are some practical limitations of the existing approaches: first, noise added through various noise distribution techniques do not considerably affect the identification of appliances; second,

the effectiveness of these approaches is quantified using information-theoretic metrics and not NILM algorithms; and finally, various auto-encoders and filters can be used to denoise a time series.

This paper solves these problems by proposing a privacy-preserving architecture that combines an obfuscator and GAN model to generate a synthetic time series that is close to the real time series. The proposed privacy preserving architecture enables a consumer to obfuscate the power consumption data to help prevent NILM algorithms from inferring the active appliances. Thus preventing consumer profiling and preserving privacy. We evaluate our approach using the widely accepted NILMKTK [7] framework and publicly available datasets such Dutch Residential Energy Dataset (DREDD) [10].

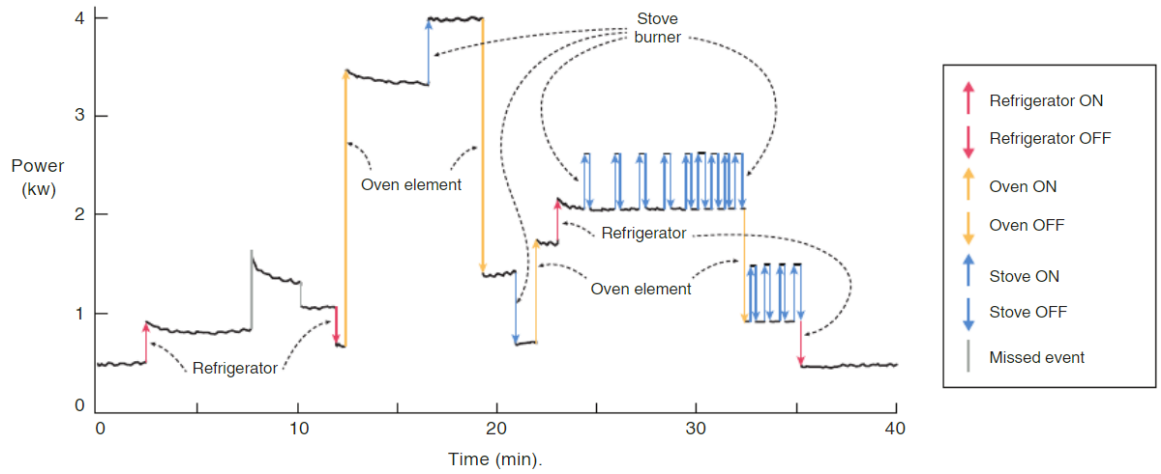

Figure 1. Appliance disaggregation from a single point of measurement of power i.e. aggregate reading using NILM algorithms to predict the appliance activity [11].

1.1. Contribution

In this paper, we propose a novel privacy-preserving architecture that will generate a synthetic time series yielding the following contributions:

1. We develop a privacy-preserving architecture to preserve the privacy of consumer activity deduced through disaggregation algorithms. The architecture identifies the inactive states and generates state combination close to the total aggregate power consumption.
2. We propose a novel hybrid privacy approach to generate indistinguishable synthetic time series data. The proposed approach hybridizes the strength of the generative adversarial network (GAN) with NILM in an adaptive manner. In this work, we develop an obfuscator model that generates the combination of the appliance's inactive state for GAN discriminator. A customized generator model is devised to produce a various robust combination of states of appliance signatures.
3. We evaluate and quantify the effectiveness of our privacy-preserving architecture by performing disaggregation on the synthetic time series generated by our architecture using NILM algorithms. We show that the disaggregation results are distinguishable from the real dataset using the MEC metric.

2. Motivation and related works

The Privacy Impact Assessment (PIA) was a comprehensive process of determining the privacy, confidentiality, and risk involved with data collection in the smart grid. Revealing information about residential consumers and activities within the house was one of the concerns reported by the privacy sub-group of the Cyber Security Working Group [12].

In 2009, the FBI Cyber intelligence investigated a wide spread incident of power theft related to the smart meters. It was found that the miscreants hacked into the smart meters and reprogrammed the power consumption settings resulting in a loss of $US 400 million annually for the Puerto Rico utility. Furthermore, in 2007, the Austin Energy/Austin Police conducted a warrantless surveillance program where consumer usage information was provided to find marijuana growing operations. Besides this, law enforcement agencies might use the data as real-time surveillance [3].

Although NILM enables efficient use of power consumption, it however, presents severe privacy concerns. The appliance usage inference from a NILM algorithm can be related to the daily routines i.e., behavioral patterns of a household or the presence of a number of individuals in a premise. Such sensitive data helps a malicious user build a detailed profile of consumer behavior in a premise and provide a basis for forecasting a premise activity such as when the premise was unattended, work schedules, and other personal activities. Furthermore, marketing agencies can use this data to carry out a targeted advertisement for devices not owned by consumers or for mass surveillance by law enforcement agencies. The potential privacy concerns and usage of data makes it a valuable target for data thieves.

Several privacy-preserving approaches have been proposed and used by researchers. We have performed an extensive literature survey on privacy-preserving schemes [3] and we have presented some state-of-the-art approaches proposed by researchers in this section.

Battery-based load hiding (BLH) approach uses a battery i.e., rechargeable, to partially supply the energy demand to manipulate meter reading to hide the actual energy consumption. [13] proposed a reinforcement learning (RL) based BLH approach to preserve privacy for high-frequency and low-frequency variation data. The RL-BLH algorithm learns a decision policy for choosing pulse magnitudes on the fly without prior knowledge of usage patterns and uses artificially generated data to reduce the time taken to converge to an optimal policy. However, reinforcement learning does not estimate the actual input/output characteristic but only the desired probabilistic behavior. [14] proposed a scheme to address the smart meter (SM) privacy concerns using renewable energy sources (RES) and a battery to partially hide the consumption pattern from the utility provider. The proposed scheme uses an information-theoretic approach to minimize leakage of consumer's energy consumption data to the utility provider as well as the energy generated by the RES. However, renewable energy is wasted when the battery is maximally charged or the required energy load is smaller than the generated energy.

Data obfuscation provides a unique opportunity to mask the original energy consumption data by applying random noise [15] or by using an appropriate algebraic transformation on the fine-grained energy usage data [16]. [17] proposed a utility-privacy tradeoff scheme based on random data obfuscation. In the proposed scheme, random data-obfuscation generated by the Laplace distribution is used to mask the real-time data. The proposed scheme also has a Key Initialization Centre (KIC) to initialize keys to smart meters and control centre and has a higher error rate. Furthermore, KIC uses Paillier encryption for generating encryption parameters, which is computationally expensive.

Data anonymization allows to disassociate the customer identity from its energy consumption data while utilities receive enough information to compute the required information. These approaches allow the implementation of additional trusted infrastructure. [18] proposed an authentication framework based on anonymization to protect unauthorized data access and achieve privacy. The framework is designed to prevent service providers from correlating various types of data from a smart meter and avoid a single point of failure. The scheme does not consider the trustworthiness concern of the Anonymizer (AN), Electricity supplier (ES), and the Data Collector (DC) colluding. [19] proposed a privacy-preserving approach based on pseudo-identity. The approach uses a hash tree-based mechanism to achieve data integrity. However, the approach does not prevent insider attacks. Furthermore, anonymization techniques have previously failed on multiple occasions [20,21], and the data was traced back to individuals.

In data aggregation, network aggregators are used for concatenating and summarizing data packets from various devices using functions such as sum or average. [22] proposed Integrated Authentication and Confidentiality (IAC) protocol to provide efficient and secure AMI communications. The scheme uses hop-by-hop data aggregation and a forwarding approach between the intermediate nodes. The proposed approach does not consider the malfunctioning of intermediate nodes and is also vulnerable attack such as replay attack and forgery attack. [23] proposed a secured privacy-preserving protocol for smart metering systems using multiple gateways for aggregation using a cluster approach. The proposed protocol uses Fully Homomorphic Encryption (FHE) with a randomly generated polynomial (secure MPC) to secure the data. The encrypted data is aggregated using a hierarchical manner and without revealing the actual meter readings. However, FHE requires a lattice-based cryptosystem, which is very complex. Thus, implementing a lattice-based cryptosystem requires significantly high and complex computations and ciphertext sizes.

Differential Privacy is another related privacy concept based on privacy-preserving data mining. The privacy mechanism adds controlled noise to the requested data before being released. [24] use the Laplace mechanism to hide the consumer's power consumption data in smart meter data sets, achieving ε-differential privacy. [25] uses a differential privacy approach by using household batteries. The battery recharges/discharges power in a bid to hide the original power consumption data. The addition of the noise depends on ε and the sensitivity function. The lower the value of ε is chosen, the privacy risk is low. However, choosing a suitable value for ε poses a difficult challenge, as it may significantly decrease the utility of the data. Furthermore, it is challenging to input a differential privacy based dataset to a complex optimization algorithm which may lose the practicality of the original power consumption dataset [26].

3. Background

In this section, we introduce the technological concept related to this work.

3.1. Generative adversarial network

GANs are deep generative models [27–29] used to produce synthetic images and text. The GAN consist of a generator (G) and a discriminator (D), which compete in a two-player min-max game V (D, G). The G learns a mapping G(z) that tries to map the random noise vector z to a realistic time series. The D tries to find a mapping D(x) that tell us the input data's probability of being real. This is

achieved by minimizing/maximizing the binary cross-entropy [30]:

$$min_G\ max_D\ V(G,D) = E_{x\ pdata(x))}[logD(x) + E_{z\ p_z(z))}[log(1 - D(G(z)))] \tag{3.1}$$

A simplified explanation of the Eq (3.1) is that the generator is trained to produce fake samples while the discriminator is trained to identify the synthetic of fake samples. A competition between the generator and the discriminator helps them improve their methods until the synthetic data is not distinguishable from the real data samples.

Algorithm 1 shows the training process for the generator model G of GANs. D and G are a neural network which try to maximize and minimize the objective, respectively. In other words, the objective of the generator G is to produce fake or synthetic data while the discriminator D is responsible for detecting the fake data samples. The feedback enables the D and G to improve their functions until the synthetic samples are indiscernible from the real data [31].

Algorithm 1 Training Algorithm of GAN

Input: Real samples { x_1, x_2 ...x_t } $p(x)$

Output: A Generative Model G

1: G ← a generative neural network
2: D ← a discriminator neural network
3: **while** until convergence of loss values **do**
4: Create mini-batch of real samples X={x_1, x_2 ...x_n}
5: Create set of latent vector inputs Z={z_1, z_2 ...z_n}
6: Train the discriminator D by maximizing equation 3.1
7: Train the generator G by minimizing equation 3.1
8: **end while**
9: **return** G

3.2. Non-intrusive load monitoring

NILM consists of machine learning algorithms that infer end-user appliances running in a consumer's premise from an aggregate power consumption obtained from a single point of source such as the smart meter. Given a smart meter, there exist an aggregate power consumption time series P = $\{p_1, p_2, \ldots p_t\}$ for time T = $\{1, 2, 3, \ldots t\}$. The NILM infers the power consumption y_t^i of appliance $i \in \{1, 2, 3 \ldots M\}$ of M active appliances such that

$$P_T = \sum_{i=1}^{M} y_t^{(i)} + \sigma(t) \tag{3.2}$$

where $\sigma(t)$ represents unaccounted power or noise.

3.3. Multi-State Energy Classifier metric

The Multi-State Energy Classifier (MEC) metric combines both event classification and energy estimation of an appliance state to give a more realistic and accurate evaluation of the performance of

the existing NILM techniques [32]. The MEC metric consist of three steps namely; calculating the classification accuracy, the energy estimation accuracy and the total penalty of the operational states of an appliance. We choose the MEC metric to measure the accuracy of the NILM algorithms for the following reasons: the MEC accurately classifies multiple states of an appliance, quantifies the accuracy even for values that are too far from the original ground truth. Also the metric does not exceed the usual accuracy interval of 0 and 1 for relatively large errors.

4. Proposed privacy preserving architecture

In this section, we introduce the security and privacy concern that will be addressed and the overall workflow of the proposed privacy-preserving architecture. We also present the hybrid-GAN, as shown in Figure 2. Figure 2 illustrates the overall architecture, which comprises of three important steps detailed in following subsections.

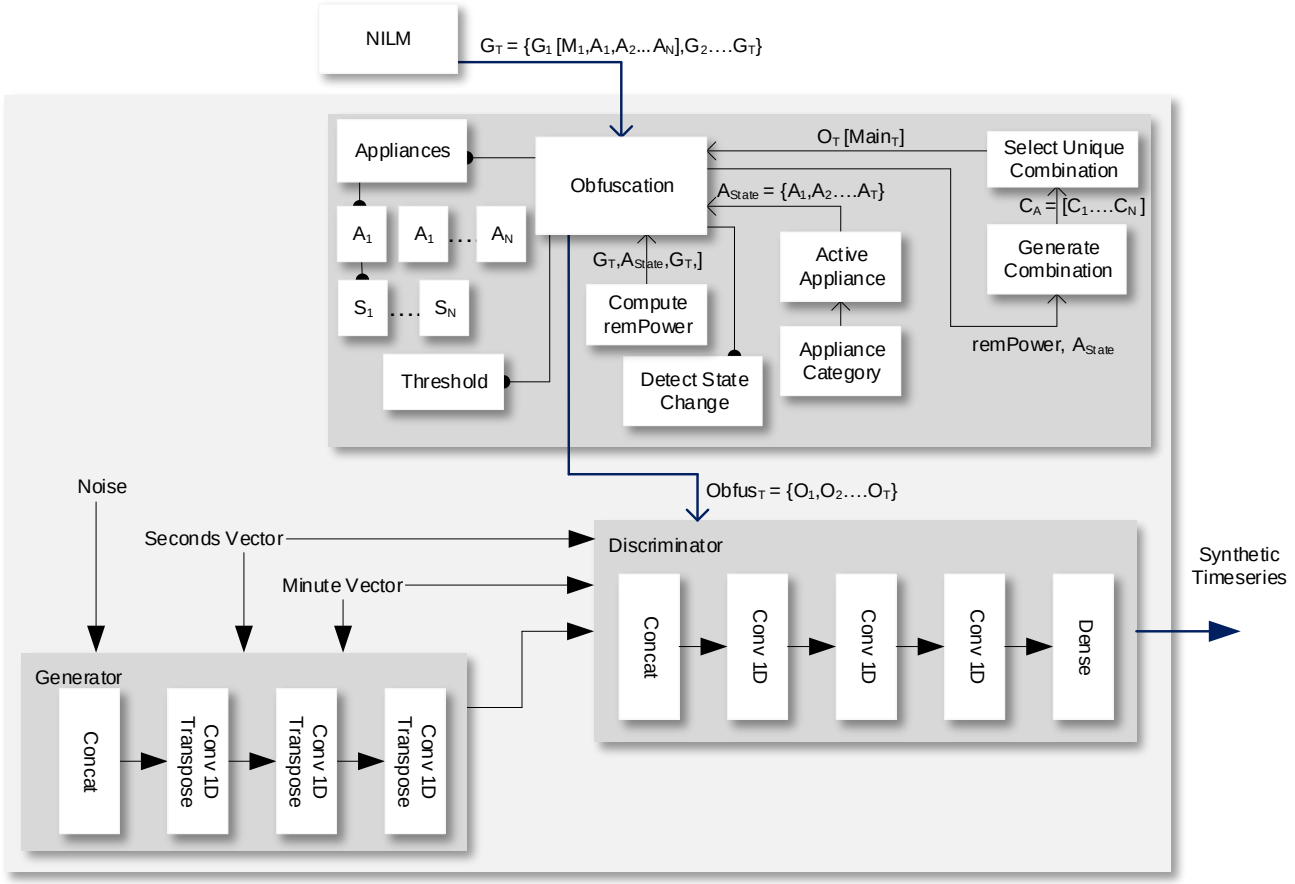

Figure 2. An overview of the proposed architecture to preserve the privacy of the energy consumption data of the consumer.

4.1. Privacy concern

We address the following privacy and security concern in this paper: infer the use of the individual appliances using a NILM algorithm from an aggregate power consumption reading i.e., consumer

profiling. Since our privacy-preserving architecture generates a synthetic time series based on the inactive states during a given time period T, it is strong against appliance inference via NILM. Unlike noise addition techniques, the hybrid-GAN effectively reduces the appliance detection accuracy and is immune to noise removing techniques such as auto-encoders and filters.

4.2. Architecture workflow

This section presents the proposed privacy-preserving architecture, as shown in Figure 2. Figure 2 illustrates the overall process, which comprises of the following steps:

1. The original power reading from the consumer's premise is disaggregated using a NILM technique and given to the data pre-processing step of the obfuscator.
2. The obfuscator process generates a combination of the appliance's inactive state. The obfuscator provides the obfuscated aggregate readings to the discriminator.
3. The GAN process in the proposed architecture is trained on the real dataset consisting of all the state combinations of the appliances used on a consumer's premises.
4. The GAN process generates a synthetic time series which is close enough to the real time series with a different combination of states of an appliance.

In the next section, we explain the obfuscator (O) in detail.

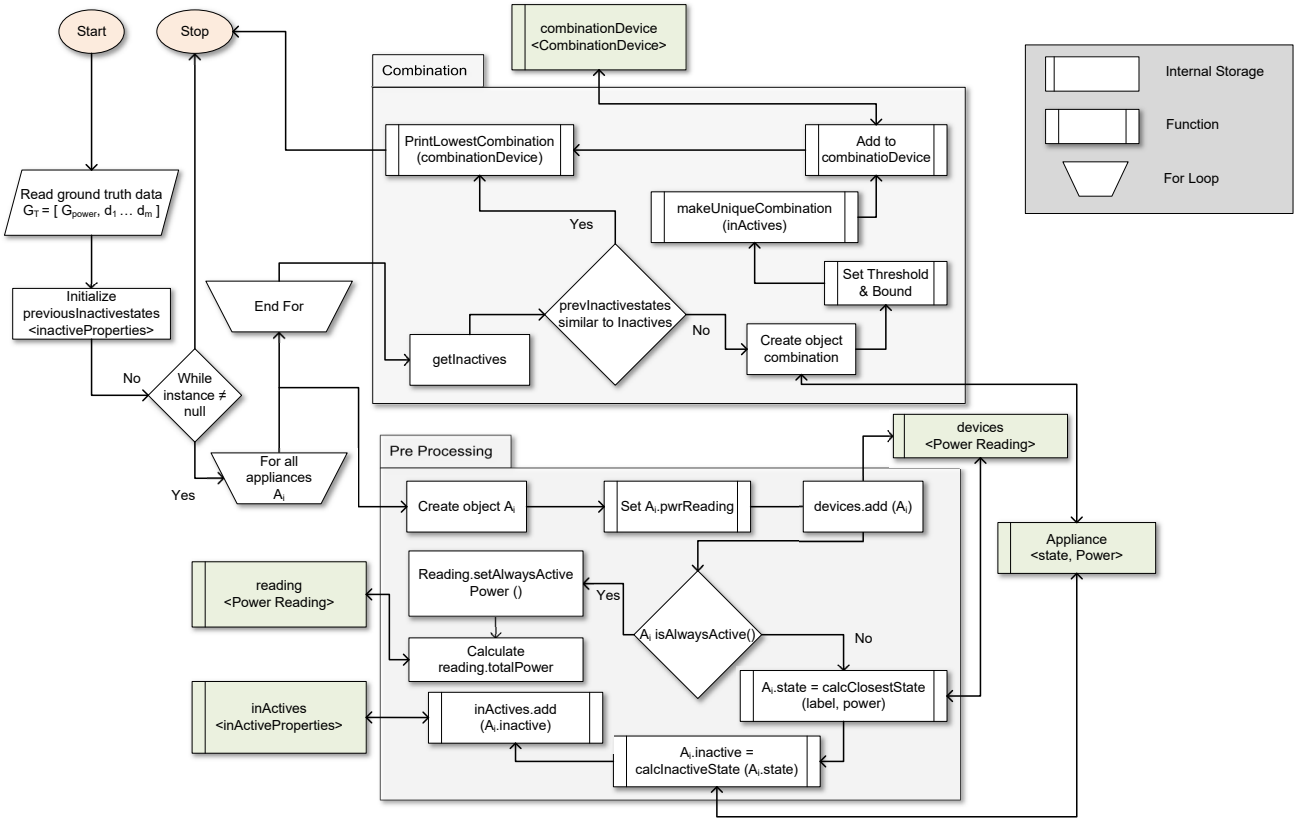

Figure 3. The diagram presents the operational flow of the proposed obfuscator module.

4.3. Obfuscator

The obfuscator process generates a combination of inactive states of appliances with total energy nearly equivalent to the original ground truth. The identification and storing of inactive states of an appliance allows the hybrid-GAN to generate n-combinations of inactive states and select the optimal solution close to the original ground truth power consumption. Figure 3 presents the obfuscation process in detail. The obfuscator takes the ground truth aggregate g_{power} and the active operational states of appliances d_i where *i=1* to m for the time T to output a obfuscated aggregate value. The obfuscator process is subdivided into two steps:

Step 1 Data pre-processing: In the data pre-processing step, the basic idea is to identify the active states and the corresponding power of the appliances, calculate the remaining power, categorize the appliances, and compute the inactive devices and its corresponding states.

The process starts by traversing through the data points of the ground truth time series $G_T = \{g_{power}, d_1, ...d_m\}$ where $g_{power} = \sum_{i=1}^{M} d_i$ at time t. An appliance object A_i is instantiated of type $< appliance >$ and the d_i power value is set to the object $A_i.power$. The A_i is stored in devices list of type $< PowerReadings >$. The process now categorizes the appliance A_i into *Always Active* or *Not Always Active*. We categorize the appliances based on the amount of privacy concern. An *Always Active* appliances such as fridge, smoke alarm do not cause privacy concerns as high as appliances in the *Not Always Active* category such as fan, television, laptop etc. These appliances help deduce a consumer's activity pattern which is a serious privacy concern. Hence we aim to obfuscate only the *Not Always Active* appliances. Based on the categorization of the appliances i.e. for an always active appliance, we set the total power of always active devices in *reading.activePower* and then calculate the remaining power to be obfuscated i.e the total power g_{power} minus the *reading.activePower*. The remaining power is then updated in *reading.remPower*.

For a *Not Always Active* appliance, the process starts by identifying the state of an appliance. The process compares the $A_i.power$ to $Map < State, Power >$ to obtain the active state of an appliance. The $Map < State, Power >$ consists of all the appliances and its states and the corresponding power of the appliance states. The states of an appliance are identified using the appliance state clustering technique as mentioned in [32]. Next, the inactive states of an appliance A_i are stored into *inActives* of type $< inActiveProperties >$. A similar process is performed for all the devices active during the same instance at time t. This is done to track the change in the state of appliances for consecutive instances.

Step 2 Generate n-device combinations: The next step in the process involves generating the state combinations for obfuscation. The *combination* process is executed only when the obfuscator detects a change in state of appliances at time $t-1$ and t. This reduces the need for re-generating the state combinations for sequentially similar active states and reduces computational time. The input for the process is the inactive states *inActives*, the total power g_{power} and a user supplied parameter *thres* as shown in Figure 4.

The *thres* is used to compute the lower bound and the upper bound for the total power of permitted combination i.e., $thres_{lower}$ and $thres_{upper}$. The process generates 'N-Appliance' combinations of all the appliances A_i and its inactive states and computes the total power of the combination. The process ensures that the appliance states of the combination i.e., $A_i.State$ are mutually exclusive. The combination is stored in *allPossibleCombination* if the total power of the combination lies within the $thres_{lower}$ and $thres_{upper}$. The process then maps the combination to the closest threshold. The N appliance combination with minimum distance is stored in *combinationDevice*. Similar process is performed for N

= 1 to number of devices in the *inActives*. At the end of the process, the *combinationDevice* is stored with best combination of N appliances. Furthermore, *combinationDevice* are sorted with respect to minimum distance and number of appliances in the combination. Once sorted, the process outputs the first combination in the *combinationDevice*.

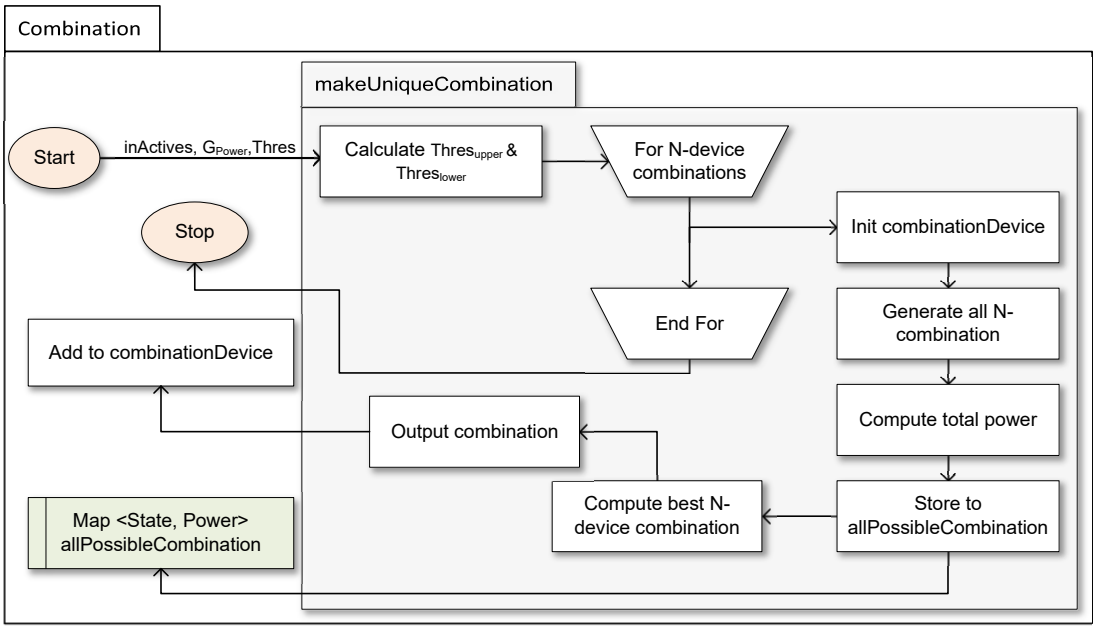

Figure 4. Process flowchart for generating mutually exclusive n-device combinations of inactive device states.

4.4. GAN architecture

In this section, we discuss the proposed GAN of the privacy-preserving architecture, as shown in Figure 2. Most of the existing GANs were developed for images process and audios applications. However, researchers have used GAN to preserve privacy of the original timeseries by generating a synthetic timeseries. In [33], author uses GAN to preserve the privacy of an individuals sensitive data generated by his movements i.e. the GPS trajectories. To this end, this work proposes a hybrid GAN architecture for the privacy preserving problems related to the consumers energy consumption data in a smart grid. The proposed architecture is designed to deal with timeseries dataset based on the model presented in [34]. To design a GAN architecture for preserving privacy in time series dataset, several changes and customisation have been made to develop an efficient and effective model. These are:

- A smart meter measures energy consumption at low-frequency sampling rate and sends data every 15 min interval, we split the time series into daily vectors i.e., minutes and seconds using hot embedding.
- We set a low decay rate to allow the GAN to have enough time to capture the pattern of the dataset.
- The aggregate power reading time series in very dynamic in nature. We use a small kernel in order to reduce the amount of inaccurate values of the power reading time series.
- We use Adam optimizer since the aggregate power time series is sparse in nature. We set the beta parameter value in order to ensure that the small weights are assigned to far gradients.

In the literature, several convolutional neural network (CNN) models were developed for image and audio classification problems. However, the same model cannot be directly used to deal with the time series data because the data is represented in sequences-time manner. CNN exhibits excellent performance on several challenging applications and thus it can be used to learn from a time series dataset. While recurrent neural networks (RNN) can also be used for timeseries data as they store temporal information available in a time series, CNN models are computationally lighter and learn by batch, in our experiments it is more suitable as we have resource constraint devices and data is sent in batches every 5–15 min (based on the utility provider) time interval. Furthermore, while RNN learns from the previous data timestep it needs to predict, whereas CNNs learns by seeing the data from a broader perspective, which is more feasible for our GAN model.

To this end, we use 1D CNN where the time series is represented as one-dimensional sequences of data. The CNN model learns to extract features from sequences of observations and then map the internal features into different activities. It directly learns most effective features from the raw time series dataset without the need for the domain expertise to extract those features. Thus the proposed model can adaptively deal with various data types and can cope with problem changes that might occur during the development process.

In this work, we develop a 1D CNN for both the discriminator (D) and Generator (G) models. We then conducted experimental tests to find the best parameter values for this GAN model. Considering the time and accuracy, we have fine-tuned various parameters by setting a small kernel size i.e. 2 and gradually increasing or decreasing the other parameters to get the best possible output. As seen in these experiments, the decay values of 1e-1 and 1e-2 decreased the learning rate rapidly for this model and resulted in poor synthetic timeseries generation. On the other hand, a smaller decay value resulted in better performance. Most of the GAN models have a default value for the beta parameter set to 0.9, but reducing it to 0.4 provided a more stable training process. Therefore, we use the suggested values as shown in Table 1.

Table 1. Fine-tuned parameters for the proposed GAN model.

Parameters	Minimum	Maximum	Suggested value
Decay Rate	1e-1	1e-7	1e-4
Beta parameter	0.1	0.9	0.4

In the following subsection, we describe the discriminator (D) and Generator (G) in detail.

4.4.1. *Discriminator*

The discriminator (D) is trained to differentiate between the generated samples as synthetic and the original samples as real. The D consists of multiple layers of a 1D CNN neural network that takes the sample input from the obfuscator, the minute vector, and the seconds vector as input and classifies whether the input is real or synthetic. The first layer of the discriminator consists of a 1D convolutional network. We set the number of filters for the layer and assign a low value for the kernel size. The kernel size specifies the size of the convolutional window. The first layer of 1D convolutional is followed by a Leaky Rectified Linear Unit (LeakyReLU) activation. We use LeakyReLU as it fixes the dying ReLU problem, is balanced and speeds up the training process. The second layer of the discriminator also

consist of a 1D convolutional network followed by batch normalization and the leakyReLU activation. The batch normalization normalizes its output with the moving average of the μ and the σ of the batch. The third layer in the discriminator is also a 1D convolutional network with batch normalization and leakyReLU. At the output, we use the sigmoid activation. The sigmoid activation exist between (0,1) and is used to predict the probability i.e., real or fake.

4.4.2. *Generator*

The generator (G) is trained to generate synthetic samples. The G consists of multiple layers of neural network that takes a latent vector Z, the minute vector, and the seconds vector as input. The first layer of the generator consists of a transpose 1D convolutional network. We set the number of filters for the layer and assign a low value for kernel size and the strides. The first layer of 1D convolutional is followed by batch normalization and the leakyReLU activation. The second layer also consist of a transpose 1D convolutional network with batch normalization and leakyReLU activation. The final layer also consists of a transpose 1D convolutional network with filters set to input dimension and followed by a sigmoid activation at the output layer.

5. Implementation & results

The implementation and results of the proposed privacy-preserving architecture for generating obfuscated timeseries is discussed in this section. The architecture is implemented in sequential order as shown in Figure 3.

5.1. Dataset description

We conduct experiments using the DREDD dataset, an open source real-world dataset for researchers [10]. We use a subset which records aggregated energy consumption and appliance level energy consumption. The aggregate power readings and the appliance level reading are collected at a sampling frequency rate of 1 Hz. The dataset consists of various appliance types such as Type I (On-Off), Type II (Multi-State) etc.

5.2. NILM algorithm

We first perform power disaggregation on the ground truth data and measure the appliance detection accuracy of three state-of-art algorithms i.e., Combinatorial Optimization (CO), Factorial Hidden Markov Model (FHMM) [7] and Sparse Viterbi [35] using the MEC metric [32]. The disaggregation algorithms are first trained on the original dataset (aggregate and appliance-level power consumption data) to identify the appliance states. Once the algorithm learns the appliance states, the algorithms are tested on the aggregate power consumption data. Table 2 presents the appliance level detection accuracy of the algorithms using the aggregate power consumption data as an input. The SparseViterbi algorithm has shown a consistent detection accuracy of more than 90% for all the appliances in the dataset. In our simulation we used four thousand (4,000) data points in a bias environment (train and test on same dataset samples).

We choose the SparseViterbi disaggregation algorithm to generate the initial input to hybrid-GAN due to the higher rate of appliance detection as shown in Table 2. The algorithm is a highly accurate

load classification and estimation algorithm. The algorithm uses a variant Viterbi algorithm and a hidden Markov model (HMM) to disaggregate appliances with complex multi-states power signatures [35].

Table 2. Detection accuracy results for state-of-art disaggregation algorithms using the MEC metric: Combinatorial Optimization (CO), Factorial Hidden markov Model (FHMM), Sparse Viterbi.

Appliance	CO	FHMM	Sparse Viterbi
Fridge	40.60%	77.28%	96.58%
Fan	29.26%	28.42%	97.02%
Cooker	19.61%	61.39%	93.72%
Television	22.47%	35.53%	97.01%
Socket	00.01%	43.03%	95.88%
Laptop	19.60%	23.32%	97.57%
Heater	03.07%	70.92%	99.59%

The algorithm outputs ground truth time series $G_T = \{g_{power}, d_1, ...d_m\}$ where $g_{power} = \sum_{i=1}^{M} d_i$ at time t where $M = 1$ to m.

5.3. Data pre-processing

In the first step, the pre-processing component is implemented on every ground truth instance at time $t = 1$ to T. In this process, the active appliances and their states are identified. The appliances are categorized into 'Always Active' and 'Not Always Active'. Based on the categorization, the remaining power to be obfuscated is calculated. The inactive states of appliances are identified as well. The output of this process is *inActives, thres, totalPower* and *remainingPower*. Table 3 presents the data pre-processing output for every instance t. The data pre-processing step calls the n-device combination process when a change of appliance states is detected, as shown in Table 3.

Table 3. A output sample of step 1 (data pre-processing) of the obfuscator of the proposed privacy-preserving architecture.

Main	**Remaining Power**	**Active States**	**State Change**	**InActive States**
149	49	Fridge (S2), Fan (S1), Laptop (S2), Heater (S1), Microwave (S1), Socket (S2)	Yes	Cooker (S1,S2), Television (S1,S2,S3), Laptop (S1,S3), Heater (S2,S3), Microwave (S2,S3), Socket (S1,S3)
149	49	Fridge (S2), Fan (S1), Laptop (S2), Heater (S1), Microwave (S1), Socket (S2)	No	No State Change
45	43	Fan (S1), Laptop (S2), Heater (S1), Microwave (S1), Socket (S1)	Yes	Cooker (S1,S2), Television (S1,S2,S3), Laptop (S1,S3), Heater (S2,S3), Microwave (S2, S3), Socket (S2,S3)
45	43	Fan (S1), Laptop (S2), Heater (S1), Microwave (S1), Socket (S1)	No	No State Change
45	43	Fan (S1), Laptop (S2), Heater (S1), Microwave (S1), Socket (S1)	No	No State Change

5.4. Generate N-device combinations

The second step of the implementation is makeUniqueCombination process. As shown in Figure 4, the process calculates the upper and lower threshold, generates n-appliance combination of inactive states. The process then selects the best optimal solution based on the minimum energy difference, the specified threshold, and the number of devices in the combination. The process outputs an obfuscated aggregate $\text{Obfus}_T = \{O_1, O_2, ...O_t\}$. Table 4 represents the ground truth and Table 5 presents the corresponding obfuscator inactive state combination output at every time instance t. The obfuscator re-calculates the states when a change in active appliances is detected in the ground truth, as shown in Table 5.

Table 4. Original ground truth data disaggregated using NILM algorithm Sparse Viterbi (Power in W).

Main Power	Fridge	Television	Fan	Laptop	Heater	Microwave	Socket	Cooker
149	98	0	29	12	2	1	7	0
149	98	0	29	12	2	1	7	0
149	98	0	29	12	2	1	7	0
45	0	0	29	12	2	0	1	0
45	0	0	29	12	2	0	1	0

Table 5. The corresponding output generated by the obfuscator against the ground truth data shown in Table 4 (Power in W).

Main Power	Fridge	Television	Fan	Laptop	Heater	Microwave	Socket	Cooker
143	98	36	0	7	2	0	0	0
143	98	36	0	7	2	0	0	0
143	98	36	0	7	2	0	0	0
41	0	0	0	7	2	25	7	0
41	0	0	0	7	2	25	7	0

5.5. GAN

The third step is the GAN to generate a synthetic time series. The discriminator, as explained in Section 4.4.1 is responsible for distinguishing between the real and synthetic data samples. The discriminator takes vector_{minute}, vector_{second} and $\text{Obfus}_T = \{O_1, O_2, ...O_t\}$ as an input. The generator, as explained in Section 4.4.2 generates synthetic data samples. The generator takes latent vector Z, vector_{minute} and vector_{second} as an input.

5.6. Discussion & results

We plot the real ground truth data and the synthetic time series output of a consumer, as shown in Figure 5 . As the obfuscator generates a combination of inactive states close to original ground truth, the energy difference is minimum. The energy difference is based on the threshold variable *thres* specified by the user to balance the utility-privacy tradeoff. We exploit the NILM feature of identifying states to make NILM predict inaccurate states. Figure 6 shows the ground truth (blue) and the disaggregated result (yellow) of generated synthetic time series for appliances i.e. fan and television. In Figure 6, at data sample t_1, the ground truth for appliance Fan is an Off state with a corresponding power of 0 Watts, whereas the NILM algorithm predicts a wrong active state with a corresponding power of 29 Watts. This shows a wrong prediction of an individual appliance activity compared to its ground truth.

Table 6 presents the accuracy scores for disaggregation of different appliances from a synthetic generated timeseries using the NILM algorithm Sparse Viterbi. We perform disaggregation on aggregate timeseries obfuscated using two approaches i.e. adding White Gaussian Noise (WGN) and our proposed method hybrid-GAN. We select appliance fridge and heater to be an always active device, while others as not always active. As mentioned before, we aim to only obfuscate the not always active devices. The results presented in table 6 show disaggregation results of hybrid-GAN based synthetic timeseries. We measure the accuracy of appliance detection rate of NILM algorithm by using the MEC [32] metric. The amount of noise added is set to 2, 3, 5 and 8% for both the approaches to show the variation between different percentage of noise levels (Min:2% - Max:8%) for our experiments. This results in a variation of 2 to 8% in the power consumption, which is acceptable in a real world scenario.

The MEC metric accurately quantifies the appliance in terms of energy estimation as well as state classification. Referring to Tables 2 and 5, the disaggregation accuracy of a SparseViterbi algorithm for the original ground truth power consumption data is 97.02, 97.01, and 97.57% for appliances Fan, Television and Laptop respectively. By adding a noise threshold of 5%, our proposed approach reduces the detection accuracy to 40.16, 39.76, and 61.60% for appliances Fan, Television and Laptop as compared to 91.21, 64.73, and 74.24% using the Gaussian noise approach. We show that our approach effectively reduces the appliance detection rate as compared to White Gaussian Noise (WGN).

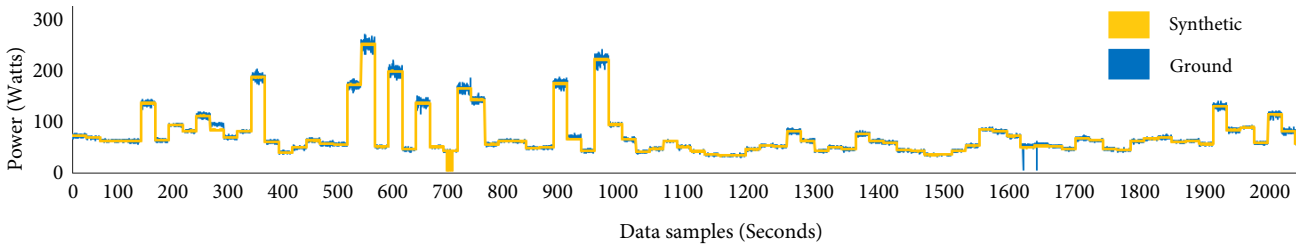

Figure 5. Ground truth (blue) and the hybrid-GAN based synthetic (yellow) aggregate time-series for a consumer.

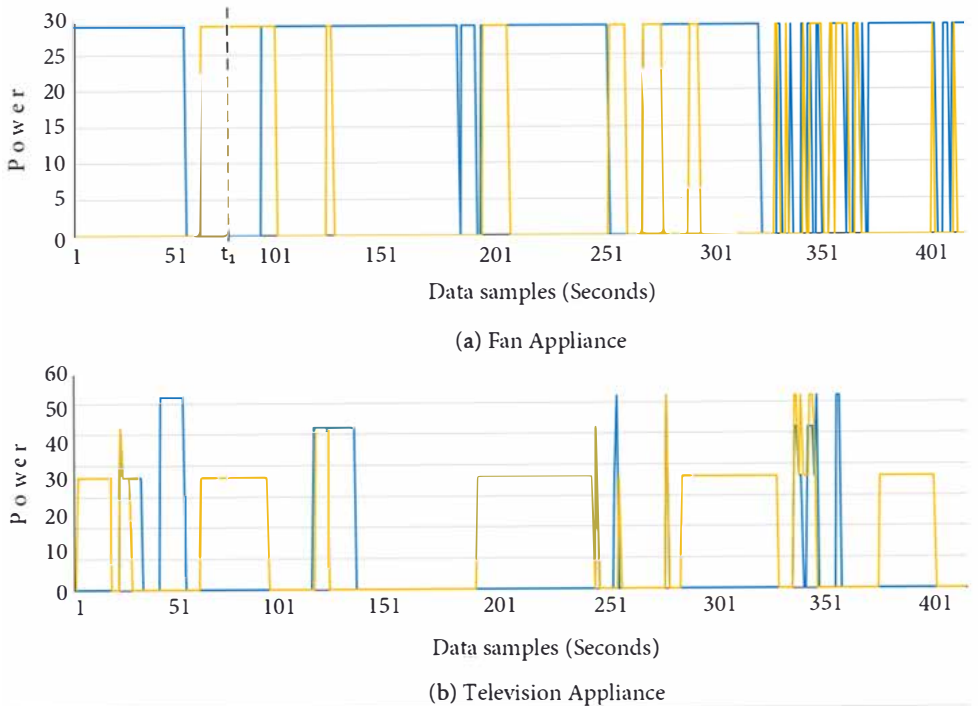

Figure 6. Ground truth (blue) and the disaggregated result (yellow) of Hybrid-GAN based synthetic timeseries for appliance Fan and Television (Power in W).

Table 6. Detection accuracy results of an aggregate synthetic timeseries generated using Gaussian Noise and Hybrid-GAN for various multi-state appliances using Sparse Viterbi and MEC.

Appliances	Gaussian Noise				Hybrid-GAN			
	2	3	5	8	2	3	5	8
Fridge	93.98%	92.71%	91.81%	91.41%	92.26%	91.21%	92.51%	91.66%
Television	74.05%	71.35%	64.73%	65.93%	49.52%	46.74%	43.17%	39.76%
Fan	90.10%	89.67%	91.21%	88.30%	81.25%	76.56%	73.04%	40.16%
Laptop	86.99%	83.38%	74.24%	72.20%	73.87%	71.69%	66.30%	61.60%
Heater	94.65%	93.72%	93.07%	94.01%	92.78%	90.29%	91.64%	88.68%
Socket	74.71%	69.41%	58.25%	47.67%	53.28%	44.13%	41.94%	36.63%

6. Conclusions & future works

This paper proposed a new privacy preserving architecture that generates a synthetic time series based on the inactive state combinations. The proposed architecture addresses the critical issues with the existing scenario: lack of effective privacy preserving approach to preserve consumer privacy and prevent inference of appliance activity. The proposed architecture solves this using a hybrid-GAN i.e. by combining the obfuscator with a generative adversarial network to generate a synthetic time series close enough to the real time series. As shown in the results, the proposed architecture has reduced

the average appliance detection accuracy of the NILM algorithm between 4-18% for devices with binary and multiple states of operation. In future works, we aim to completely integrate the obfuscator as part of the GAN. This will enable GAN to generate specific combinations corresponding to the aggregate power based on constraints and conditions. Furthermore, we also aim to include appliance state selection based on time of the day and the appliance in use relating to the consumer. For example, using a BBQ appliance in early hours of morning would notify a malicious user of a synthetic time series in use. Synchronizing the appliance and time of use will help mislead malicious user as the inactive state combinations will be more time-based and will reflect a normal consumer activity when analysed.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. US Energy Information Administration, *Internation energy outlook 2019*. Available from: https://www.eia.gov/outlooks/ieo/pdf/ieo2019.pdf.
2. International Energy Agency, *How to Guide for Smart Grids in Distribution Networks Roadmap Development and Implementation*. Available from: https://www.ctc-n.org/sites/www.ctc-n.org/files/resources/technologyroadmaphow2guideforsmartgridsindistributionnetworks.pdf.
3. S. Desai, R. Alhadad, A. Mahmood, N. Chilamkurti, A survey of privacy preserving schemes in ioe enabled smart grid advanced metering infrastructure, *Cluster Comput.*, **22** (2019), 43–69. https://doi.org/10.1007/s10586-018-2820-9
4. Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. Kalogridis, M. Sooriyabandara, et al., Smart grid communications: Overview of research challenges, solutions, and standardization activities, *IEEE Commun. Surv. Tutorials*, **15** (2013), 21–38. https://doi.org/10.1109/SURV.2011.122211.00021
5. J. Zhang, Z. Chen, X. Yang, K. Chen, K. Li, Ponder over advanced metering infrastructure and future power grid, in *2010 Asia-Pacific Power and Energy Engineering Conference*, (2010), 1–4. https://doi.org/10.1109/APPEEC.2010.5448797
6. K. C. Armel, A. Gupta, G. Shrimali, A. Albert, Is disaggregation the holy grail of energy efficiency? the case of electricity, *Energy Policy*, **52** (2013), 213–234. https://doi.org/10.1016/j.enpol.2012.08.062
7. N. Batra, J. Kelly, O. Parson, H. Dutta, W. Knottenbelt, A. Rogers, et al., Nilmtk: an open source toolkit for non-intrusive load monitoring, in *Proceedings of the 5th international conference on Future energy systems*, (2014), 265–276. https://doi.org/10.1145/2602044.2602051
8. E. Ireland, Puerto rico smart meters believed to have been hacked – and such hacks likely to spread, 2012. Available from: https://www.smart-energy.com/regional-news/north-america/puerto-rico-smart-meters-believed-to-have-been-hacked-and-such-hacks-likely- to-spread/.
9. R. Guest, Austin police- it's not illegal, but we won't disclose it, 2007. Available from: https://www.dallascriminaldefenselawyerblog.com/austin-police-its-not-illegal/.

10. A. S. N. U. Nambi, A. R. Lua, V. R. Prasad, Loced: location-aware energy disaggregation framework, in *Proceedings of the 2Nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, (2015), 45–54. https://doi.org/10.1145/2821650.2821659

11. L. Pereira, N. Nunes, Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools—a review, *WIREs Data Min. Knowl. Discovery*, **8** (2018), e1265. https://doi.org/10.1002/widm.1265

12. V. Y. Pillitteri, T. L. Brewer, *Guidelines for Smart Grid Cybersecurity*, 2014. https://doi.org/10.6028/NIST.IR.7628r1

13. J. Koo, X. Lin, S. Bagchi, Rl-blh: learning-based battery control for cost savings and privacy preservation for smart meters, in *2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, (2017), 519–530. https://doi.org/10.1109/DSN.2017.16

14. G. Giaconi, D. Gündüz, Smart meter privacy with renewable energy and a finite capacity battery, in *2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, (2016), 1–5. https://doi.org/10.1109/SPAWC.2016.7536745

15. O. Vuković, G. Dán, R. B. Bobba, Confidentiality-preserving obfuscation for cloud-based power system contingency analysis, in *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, (2013), 432–437. https://doi.org/10.1109/SmartGridComm.2013.6687996

16. A. R. Borden, D. K. Molzahn, P. Ramanathan, B. C. Lesieutre, Confidentiality-preserving optimal power flow for cloud computing, in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, (2012), 1300–1307. https://doi.org/10.1109/Allerton.2012.6483368

17. Z. Guan, G. Si, J. Wu, L. Zhu, Z. Zhang, Y. Ma, Utility-privacy tradeoff based on random data obfuscation in internet of energy, *IEEE Access*, **5** (2017), 3250–3262. https://doi.org/10.1109/ACCESS.2017.2662940

18. S. Afrin, S. Mishra, An anonymized authentication framework for smart metering data privacy, in *2016 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, (2016), 1–5. https://doi.org/10.1109/ISGT.2016.7781185

19. H. Bao, L. Chen, A lightweight privacy-preserving scheme with data integrity for smart grid communications, *Concurrency Comput.: Pract. Exper.*, **28** (2016), 1097–1110. https://doi.org/10.1002/cpe.3527

20. A. Narayanan, V. Shmatikov, Robust de-anonymization of large sparse datasets, in *2008 IEEE Symposium on Security and Privacy (sp 2008)*, (2008), 111–125. https://doi.org/10.1109/SP.2008.33

21. L. Sweeney, K-anonymity: A model for protecting privacy, *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, **10** (2002), 557–570. https://doi.org/10.1142/S0218488502001648

22. Y. Yan, R. Q. Hu, S. K. Das, H. Sharif, Y. Qian, An efficient security protocol for advanced metering infrastructure in smart grid, *IEEE Network*, **27** (2013), 64–71. https://doi.org/10.1109/MNET.2013.6574667

23. S. Tonyali, K. Akkaya, N. Saputro, A. S. Uluagac, M. Nojoumian, Privacy-preserving protocols for secure and reliable data aggregation in iot-enabled smart metering systems, *Future Gener. Comput. Syst.*, **78** (2018), 547 – 557. https://doi.org/10.1016/j.future.2017.04.031
24. G. Ács, C. Castelluccia, I have a dream! (differentially private smart metering), *Int. Workshop Inf. Hiding*, (2011), 118–132. https://doi.org/10.1007/978-3-642-24178-9_9
25. J. Zhao, T. Jung, Y. Wang, X. Li, Achieving differential privacy of data disclosure in the smart grid, in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, (2014), 504–512. https://doi.org/10.1109/INFOCOM.2014.6847974
26. F. Fioretto, T. W. K. Mak, P. V. Hentenryck, Differential privacy for power grid obfuscation, *IEEE Trans. Smart Grid*, **11** (2020), 1356–1366. https://doi.org/10.1109/TSG.2019.2936712
27. D. Jeong, B. Kim, S. Dong, Deep joint spatiotemporal network (djstn) for efficient facial expression recognition, *Sensors*, **20** (2020), 1936. https://doi.org/10.3390/s20071936
28. J. Kim, B. Kim, P. P. Roy, D. Jeong, Efficient facial expression recognition algorithm based on hierarchical deep neural network structure, *IEEE Access*, **7** (2019), 41273–41285. https://doi.org/10.1109/ACCESS.2019.2907327
29. Y. Yuan, J. Chu, L. Leng, J. Miao, B. Kim, A scale-adaptive object-tracking algorithm with occlusion detection, *J. Image Video Proc.*, 2020. https://doi.org/10.1186/s13640-020-0496-6
30. Y. Luo, X. Cai, Y. Zhang, J.Xu, X. Yuan, Multivariate time series imputation with generative adversarial networks, in *32nd Conference on Neural Information Processing Systems*, Montréal, Canada, (2018), 1596–1607.
31. N. Park, M. Mohammadi, K. Gorde, S. Jajodia, H. Park, Y. Kim, Data synthesis based on generative adversarial networks, *Proc. VLDB Endowment*, **11** (2018), 1071–1083. https://doi.org/10.14778/3231751.3231757
32. S. Desai, R. Alhadad, A. Mahmood, N. Chilamkurti, S. Rho, Multi-state energy classifier to evaluate the performance of the nilm algorithm, *Sensors*, **19** (2019), 5236. https://doi.org/10.3390/s19235236
33. X. He, G. Cormode, A. Machanavajjhala, C. M. Procopiuc, D. Srivastava, Dpt: Differentially private trajectory synthesis using hierarchical reference systems, *Proc. VLDB Endowment*, **8** (2015), 1154–1165. https://doi.org/10.14778/2809974.2809978
34. A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, preprint, arXiv:1511.06434.
35. S. Makonin, F. Popowich, I. V. Bajić, B. Gill, L. Bartram, Exploiting hmm sparsity to perform online real-time nonintrusive load monitoring, *IEEE Trans. Smart Grid*, **7** (2016), 2575–2585. https://doi.org/10.1109/TSG.2015.2494592

AIMS Press