Electricity Theft Detection Base on Extreme Gradient Boosting in AMI

Zhongzong Yan, *Student Member, IEEE*, and He Wen, *Senior Member, IEEE*

***Abstract*— Metering data from the advanced metering infrastructure can be used to find abnormal electricity behavior for the detection of electricity theft, which causes huge financial losses to electric companies every year. This article proposes an electricity theft detector using metering data based on extreme gradient boosting (XGBoost). The metering data are preprocessed, including recover missing and erroneous values and normalization. The classification model based on XGBoost is trained using both benign and malicious samples. Simulations are done by using the Irish Smart Energy Trails data set with six certain attack types. Compared with the support vector machine, decision tree, and other eight machine learning methods, the proposed method can detect electricity theft with either higher accuracy or lower false-positive rate. Experiment results also demonstrate that the proposed method is robust when the data are imbalanced. Our codes are available at https://github.com/WenHe-Hnu/Electric_Theft_XGBoost.**

***Index Terms*— Advanced metering infrastructure (AMI), electricity theft, extreme gradient boosting (XGBoost), metering data, nontechnical loss.**

I. Introduction

ADVANCED metering infrastructure (AMI), as one of the key components of a smart grid, consisting of smart meters that monitor the power usage, communicate, and control to optimize the energy usage, operate data management systems to store, and process metering and control data [1]. The real-time monitoring and control of smart meters are the essential steps toward the future smart grid [2]. However, smart meters also introduce numerous new methods of electricity theft. Malicious users can hack into smart meters by using advanced instruments or cyberattack techniques [3]. This illegal practice causes a huge amount of financial loss every year. Practically, all electric companies around the world, particularly those in many developing countries [4], suffer from the adverse effects of electricity theft. Financial loss due to electricity theft in the US, Canada, and India was about $6 billion [5], 100 million Canadian dollars [6], and $17 billion per year [7], respectively.

Manuscript received July 26, 2020; revised November 20, 2020; accepted December 13, 2020. Date of publication January 1, 2021; date of current version January 21, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61771190 and in part by the Hunan Provincial Natural Science Foundation of China under Grant 2019JJ20001. The Associate Editor coordinating the review process was Peter Xiaoping Liu. *(Corresponding author: He Wen.)*

The authors are with the Hunan Province Key Laboratory of Intelligent Electrical Measurement and Application Technology, College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: yanzhongzong@163.com; he_wen82@126.com).

Digital Object Identifier 10.1109/TIM.2020.3048784

Traditionally, technicians must analyze monthly metering data collected over a long period of time to find the abnormal use of electricity and then they go to each resident community in person to check the status and connection of each meter [8]. This approach heavily depends on the knowledge of experts, which is difficult to handle the massive real-time electricity consumption data. In addition, the judgment made by experts is limited because the data that require analysis are increasing day-to-day [9].

The research of machine learning methods provides a new prospect for electric companies to recognize abnormal electricity consumption patterns from massive data. These methods can alleviate the burden of technicians and improve detection accuracy by identifying abnormal patterns. Thus, the high-performance classifier is needed to assist the existing methods to deal with heavy detection tasks.

To combat electricity theft, many methods have been proposed in the literature in AMI in recent years. They can be divided into three groups [10].

1) State-based approaches [11]–[15] use specific instruments or design some kind of metering devices to address the issues of electricity theft. For example, in [11], an adapted ammeter is designed for fraud detection in low-voltage installations, where the operator can check and compare the difference between the local and remote measures to detect electricity theft. However, electricity theft detection based on state estimation can only be used at the substation level instead of at the end-user level. In addition, additional devices will cost extra, and some kinds of devices are difficult to install in the existing distribution network.

2) Game theory-based models [16], [17] consider these tampering behaviors to be a game between malicious users and electric companies. The goal of this type of method is to find the Nash equilibrium for this game. The game theory-based methods are relatively less expensive, but they are hard to find a suitable equation to explain the relationship between users and electric companies.

3) Artificial intelligence (AI)-based methods [18]–[21] are the most commonly used in AMI, which exploits machine learning techniques to analyze the load profiles and electricity consumption pattern of users to find suspicious users of electricity theft. Classification methods usually involve using the users' historical electricity usage data with labels to find abnormal patterns and then to detect potential electricity theft behaviors. In contrast, clustering methods [22] focus on the information without labels, i.e., the outliers are found by analyzing the relationship among users.

However, many AI-based methods have low accuracy for some specific reasons. Neural networks, for example, are prone to overfitting because it learns the training samples very well but fails to generalize the new samples. Other methods, such as SVM, show poor performance when the data set is distorted by noise [20] (e.g., missing values). In addition, the performance of shallow structure models is also limited by the high-dimensional data [8]. Thus, it is important to design an efficient model for electricity theft detection to tackle those limitations.

Recently, extreme gradient boosting (XGBoost) has been proposed by [23], which is a scalable end-to-end tree boosting system. XGBoost is an implementation of the gradient boosting decision tree (DT) algorithm. It has several advantages, such as adding regularization and exploiting out-of-core computing for very large data sets that do not fit into memory. In addition, XGBoost uses parallel and distributed computing to speed up the process of learning, enabling quicker model exploration.

This article represents the extension of the research activity presented in [24], which proposes an electricity theft detector using metering data based on XGBoost. The proposed XGBoost-based electricity theft detector has two phases. In the training phase, benign metering data are first converted to the proper format. The second step of the training phase is data preprocessing, including recover missing or erroneous values and normalization. Benign samples are obtained once the data preprocessing is completed. Malicious samples are generated by modifying benign samples according to attack types [19]. The final step is training the classification model based on XGBoost using both benign and malicious samples. In the application phase, the trained classification model is applied to new samples to determine whether they belong to a benign or malicious class. Simulations are done by using the Irish Smart Energy Trails data set [25]. Compared with the support vector machine (SVM), backpropagation neural network (BPNN), extreme learning machine (ELM), deep ELM (DELM), logistic regression (LR), DT, random forest (RF), AdaBoost, Naïve Bayes (NB) algorithm, and k-nearest neighbors (kNNs) algorithm, the proposed method can detect electricity theft with either higher accuracy or lower false-positive rate (FPR). Experiment results also show that the proposed method presents an excellent performance in electricity theft detection when the data set is imbalanced.

The remainder of this article is structured as follows. In Section II, the AMI architecture and attack models are introduced, and the outcome characterization of electricity theft and the process of the proposed method are also shown in this section. The relative techniques used in the proposed method, such as data preprocessing, classification and regression tree (CART), and XGBoost, are shown in Section III. The simulation methodology and experiment results analysis are presented in Section IV. Section V gives the conclusion and shows the limitations of the proposed method.

II. Problem Description

In this section, the architecture of AMI and the attack models that are considered in this article are briefly introduced.

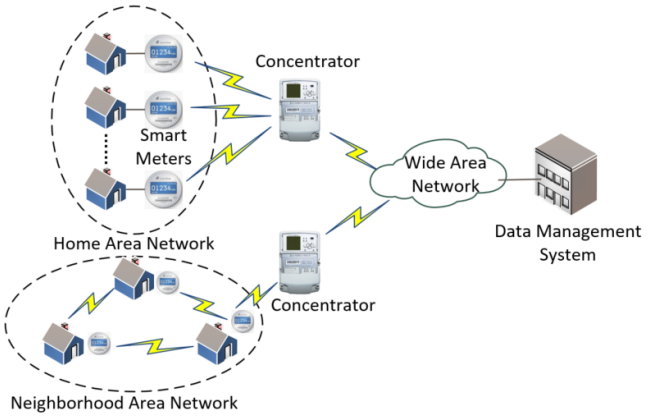


Fig. 1. Simplified architecture of AMI.

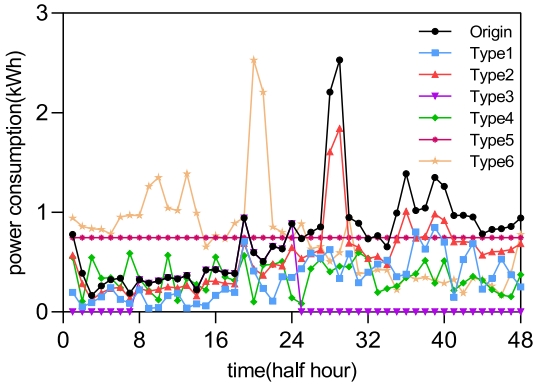


Fig. 2. Example of a day consumption and attack patterns.

Then, we analyze the characterization of electricity theft theoretically. Finally, the process steps of the proposed method are presented.

A. AMI System Model

The architecture of AMI is illustrated in Fig. 1. AMI is composed of smart meters, communications networks, and data management system. As can be seen from Fig. 1, a concentrator is installed in an area with a group of smart meters. Each customer is equipped with a smart meter that connects to the smart devices at home to aggregate their energy consumption [26].

A concentrator collects and records the amount of metering data from a home area network or neighborhood area network. With the help of these equipment and communication networks, the data management system can monitor the real-time electricity consumption data remotely, which effectively improves the detection ability and response ability.

B. Attack Models

There are many known techniques for electricity theft, which can be categorized into three groups [27].

1) *Physical Attacks:* Malicious users manipulate smart meters physically to lower meter readings, such as bypassing meters and tampering with the meters.

TABLE I
SIX TYPES OF MALICIOUS SAMPLES

Attack Types	*Modification*
Type 1	$\tilde{x}_t = \alpha x_t,\ 0.2 < \alpha < 0.8$
Type 2	$\tilde{x}_t = \alpha_t x_t,\ 0.2 < \alpha_t < 0.8$
Type 3	$\tilde{x}_t = \beta x_t,\ \beta = \begin{cases} 1 & \text{if } t_1 < t < t_2 \\ 0 & \text{otherwise} \end{cases}$
Type 4	$\tilde{x}_t = \alpha_t \bar{\mathbf{x}},\ 0.2 < \alpha_t < 0.8$
Type 5	$\tilde{x}_t = \bar{\mathbf{x}}$
Type 6	$\tilde{x}_t = x_{48-t}$

2) *Cyberattacks:* Malicious users compromise meter readings remotely or modify the firmware on smart meters using communication technologies.
3) *Data Attacks:* Malicious users inject bad data into the data management system or smart meters, which reduces their electricity bills and meter readings.

In this article, we focus on the detection of malicious users who report abnormal metering data and assume that they cannot manipulate other users' meters.

Furthermore, six attack types proposed by Jokar *et al.* [19] are used to simulate the acts of malicious users. The smart-metering data set used in this article is provided by the Irish Smart Energy Trail data set [25]. The purpose of the trial is to evaluate the performance of smart meters and analysis the impact on the system load profile, which provides the basis for widespread use of smart meters. This data set collects the metering data of more than 5000 Irish household and business customers for over 500 days between 2009 and 2010.

There are six files in this data set, and each file contains 533 days' metering data for each customer, which is recorded every half hour. This means that the daily metering data of each customer can be represented by a vector containing 48 components. We assume that all the historical data have not been manipulated by malicious users.

Denote $\mathbf{x} = [x_1, x_2, \ldots, x_{48}]$ as the users' metering data in a day (i.e., smart meters send metering data (in Watt) to data management system every 30 min). In order to generate malicious samples, we use the methods proposed by Jokar *et al.* [19] to generate six attack types to modify metering data. The details of how to generate these six attack types are shown in Table I.

As shown in Table I, Type 1 represents that all meter readings multiplied the same randomly generated parameter α, which ranges from 0.2 to 0.8. Type 2 represents each meter reading multiplied by a different random number α_t. Type 3 means that the smart meter sends its meter reading during the time period $t_1 - t_2$ and sends zero for the other period, where $t_1 - t_2$ is a randomly defined time period longer than 6 h. Type 5 indicates that smart meters send the mean value of metering data over the day to the data management system and type 4 multiply by a random factor α_t on the basis of Type 5. Type 6 means that malicious users reverse the order of their meter readings during the day. Fig. 2 shows an example of normal daily electricity consumption pattern and six malicious attack models.

C. Problem Analysis

Electricity theft refers to the behavior of illegal use of electric energy. For the purpose of not paying or paying fewer electricity bills, illegal users use different approaches to reduce the meter readings. In this article, we consider the topology of the distribution network as radial, which can be represented as the tree shape topology where close loops do not exist. In this type of network, each smart meter connects to the same concentrator.

In this article, we assume that there is no delay in sending and receiving data between the smart meter and the concentrator. Thus, in a neighborhood area network, the metering data during the time period t in a concentrator can be represented as

$$E_t = \sum_{i=1}^{N} q_t^{(i)} + \eta + \delta \tag{1}$$

where E_t represents the metering data of a concentrator during a time period t. $q_t^{(i)}$ represents the reported metering data of the user i during a time period t. η and δ represent technical losses (TLs) and measurement error, respectively.

We assume that there are N users in a neighborhood area network, and the concentrator receives meter reading from smart meters automatically in a time interval t. In this article, we do not focus on the calculation of TLs and assume that all smart meters have no errors. Thus, we denote the discrepancy of metering data between the concentrator and smart meters as the reduced meter readings due to electricity theft and is computed as $\Delta q_t = E_t - \sum_{i=1}^{N} q_t^{(i)}$. If Δq_t larger than a threshold value in continuous days, we call those smart meters failing to meet the balance check and there may be electricity theft in this neighborhood area network.

Denote q_i as the actual amount of electricity consumed by customer i and $\tilde{q}_i$ as the meter reading of customer i. Thus, the discrepancy between the actual electricity consumption data and meter readings in a specific area that includes n users can be represented as

$$\Delta q = \sum_{i=1}^{n} \left(q^{(i)} - \tilde{q}^{(i)}\right). \tag{2}$$

The goal of malicious users is to maximize Δq and minimize the likelihood of being detected, i.e., the meter readings data of malicious users are less than the amount of electricity that they actually consumed. Therefore, the purpose of the proposed method is to identify malicious users by constructing a classifier model using meter reading data. The proposed method is based on the following steps.

1) Each user contains half-hourly reported meter readings for 533 days. This means that each user has 533 vectors of 48 features to describe the daily meter reading data. Then, anomaly values or missing values are processed by the mean value method and the interpolation method.
2) For each user, we randomly select 50% of the samples to generate six malicious models through the abovementioned methods. We choose 80% and 20% of the samples as the training set and the test set, respectively.

3) We build the classification model and train the classifier with the training set and compared the results with the test set.

The details of these steps are presented in the following.

III. Methodology

In this section, we introduce the XGBoost-based electricity theft detector, which detects electricity theft by tracking user historical metering data and electricity usage data.

A. Data Preprocess

Meter reading data may contain erroneous data or missing data, which may be due to a variety of reasons, such as the broken electrical components, aging of resistance, the error of transmission, and bad connections. In the proposed model, the mean value method is used to recover these missing data, and the equation is presented as follows:

$$f(x_i) = \begin{cases} \text{mean}(\mathbf{x}_i) & x_i \in \text{NaN} \\ x_i & \text{otherwise} \end{cases} \tag{3}$$

where $\text{mean}(\mathbf{x}_i)$ is the average of vector $\mathbf{x}_i$; x_i is the value of the electricity consumption data in one cycle (i.e., half an hour). Denote NaN as if x_i is not a number value.

In addition, there are also some erroneous values (i.e., outliers) in the metering data [20]. Therefore, we recover those values by the following equation according to the "three-sigma rule of thumb":

$$f(x_i) = \begin{cases} \dfrac{x_{i-1}+x_{i+1}}{2} & \text{if } x_i > 3\cdot\sigma(\mathbf{x}_i) \text{ and } x_{i-1}, x_{i+1} \neq \text{NaN} \\ x_i & \text{otherwise} \end{cases} \tag{4}$$

where $\sigma(\mathbf{x}_i)$ represents the standard deviation of vector $\mathbf{x}_i$.

B. CART

The CART is a very popular tree-based algorithm that is widely used in machine learning [28]. The CART algorithm uses a new metric called the Gini index to select the partitioning attribute. In the classification task, the Gini index of the probability distribution is defined as

$$\text{Gini}(p) = \sum_{k=1}^{K} p_k(1 - p_k) \tag{5}$$

where K represents the number of classes and p_k represents the probability of sample belongs to the kth class.

For the binary classification problem, the Gini index of a probability distribution can be represented as

$$\text{Gini}(p) = 2p(1 - p). \tag{6}$$

Denote the Gini index of the given sample set D as

$$\text{Gini}(D) = 1 - \sum_{k=1}^{K} \left(\frac{|C_k|}{D}\right)^2 \tag{7}$$

where C_k represents the subset belonging to the kth class.

The sample set D is divided into subsets D_1 and D_2 according to whether feature A equals to a possible value a. D_1 and D_2 are defined as

$$\begin{aligned} D_1 &= \{(x, y) \in D | A(x) = a\} \\ D_2 &= D - D_1. \end{aligned} \tag{8}$$

Denote the Gini index of sample set D under feature A as

$$\text{Gini}(D, A) = \frac{|D_1|}{|D|}\text{Gini}(D_1) + \frac{|D_2|}{|D|}\text{Gini}(D_2) \tag{9}$$

where $\text{Gini}(D)$ represents the uncertainty of the sample set D, and $\text{Gini}(D, A)$ indicates the uncertainty of sample set D after $A = a$ partition.

Given the sample set D, the CART algorithm calculates the Gini index of each feature on the sample set. For each feature A and its possible value a, the Gini index is calculated when $A = a$ and splitting the sample set D into subset D_1 and D_2 according to feature A equal to a or not. For all features A and its possible value a, the feature with the smallest Gini index and its corresponding segmentation point will be used to generate two child nodes from the current node, and the training set is distributed to the two child nodes according to the optimal feature. Recursively implement the above process until meeting the stop condition.

C. Gradient Tree Boosting

XGBoost [23] is a scalable tree boosting system, mainly used to solve supervised learning tasks. It refers to learning in which we train the machine using training data $\mathbf{x}_i$ that are well labeled to predict the target variable $\hat{y}_i$.

Given a classification task, a data set D can be represented as

$$D = \{(\mathbf{x}_i, y_i)\}\big(|D| = n,\ x \in \mathbb{R}^m, y_i \in \mathbb{R}\big) \tag{10}$$

where $\mathbf{x}_i$ is a vector with n samples and m features, and y_i is the label. In this article, $\mathbf{x}_i$ denotes the meter reading for a day period with n samples and $\mathrm{y}_i = \{0, 1\}$, where $y_i = 0$ indicates that the user has no abnormal power usage behavior and vice versa.

The output of a tree ensemble model using K additive functions f_k is represented as

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i), \quad f_k \in F \tag{11}$$

where F is the space of functions containing all classification trees.

Instead of learning the weights in the tree model, XGBoost is learning functions. The objective function of XGBoost is presented as follows:

$$\text{Obj} = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \text{ with } \Omega(f) = \gamma T + \frac{1}{2}\lambda\|w^2\| \tag{12}$$

where l is the loss function, which measures how well the model fits on training data (i.e., the difference between prediction $\hat{y}_i$ and target y_i). Ω is the regularization term that

measures the complexity of the model. XGBoost adds to L_1 and L_2 regularization terms on the base of gradient boosting DT. T represents the number of leaf nodes, and w represents the scores of leaf nodes. The benefit of regularization term helps to prevent overfitting.

The model is trained in an additive manner. Denote $\hat{y}_i^{(t)}$ as the prediction term of the ith instance at the tth iteration. The objective of the ith instance at the tth iteration can be optimized

$$L^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t). \tag{13}$$

Taking the Taylor expansion of the objective

$$L^{(t)} \approx \sum_{i}^{n} \left[l\left(y_i, \hat{y}^{(t-1)}\right) + g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_t) \right] + \Omega(f_t) \tag{14}$$

where g_i and h_i are the first- and second-order gradient stochastics on the loss function, respectively

$$g_i = \partial_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right) \tag{15}$$

$$h_i = \partial^2_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right). \tag{16}$$

Denote $I_j = \{i | q(\mathbf{x}_i) = j\}$ as the instance set of leaf j. After removing all the constant terms and expanding Ω, the specific objective at step t becomes

$$\begin{aligned} L^{(t)} &= \sum_{i=1}^{n} \left[g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i) \right] + \Omega(f_t) \\ &= \sum_{i=1}^{n} \left[g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i) \right] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \\ &= \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda w_j^2 \right) \right] + \gamma T. \end{aligned} \tag{17}$$

Equation (17) becomes the optimization goal for the new tree, and this is how XGBoost supports custom loss functions. The main advantage of XGBoost is that it adds regularizations to the loss function, which makes the new tree simpler and prevents overfitting, while it does not reflect the speed advantage of XGBoost. One of the reasons why XGBoost is fast is that the column block is implemented in engineering, making parallel training possible. More details about how XGBoost works are given in [23].

IV. Evaluations

In this article, we conduct all experiments in a PC with an i7-8550U CPU and 16-GB RAM, and the programming work is done on PyCharm 2019. The evaluation metrics used to evaluate the proposed method are precision, FPR, recall, and AUC. The parameters of XGBoost are set as follows: the booster is gbtree; the L_2 regularization term on weights is 2; the learning rate is 0.05; and the maximum depth of trees is 8. Our codes are available at https://github.com/WenHe-Hnu/Electric_Theft_XGBoost.

TABLE II
Confusion Matrix Applied in Electricity Theft Detection

Data samples	Predict as malicious	Predict as benign
Malicious actually	True positives (TP)	False negatives (FN)
Benign actually	False positives (FP)	True negatives (TN)

To demonstrate the effectiveness of the proposed method, we compare the proposed method with ten AI-based models from seven types, and these schemes are presented as follows.

1) *SVM [19]:* It is one of the most used models in electricity theft detection. It separates the different classes by constructing a hyperplane. The parameters of SVM are set as $C = 50$, and the kernel function is the radial basis function.
2) *Neural Network-Based Methods:* Three neural networks are used for comparison in this experiment: BPNN [18], ELM [29], and DELM. It is worth noting that the parameters of BPNN are set as follows: the number of hidden layers is one, and the numbers of neurons in the hidden layer and input layer are 48. The neuron of the output layer is one. The network structure of ELM is the same as BPNN, and the number of hidden layers in DELM is 48.
3) *DR [30]:* It is a supervised learning algorithm that builds a tree structure by splitting a data set into smaller sets. Each branch represents the outcome of the test, and each leaf node represents a class label.
4) *kNN [31]:* It is one of the simplest algorithms used to classify the new data based on a similarity measure. The parameters are set as $k = 5$ and $p = 2$.
5) *LR:* It is a linear algorithm for a binary classification problem. It is used to predict a binary outcome given a set of independent variables. The hyperparameter of LR is set as $C = 1$.
6) *NB [32]:* It is a probabilistic algorithm that is used for classification problems. It assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.
7) *Ensemble Methods:* We compare the proposed method with two ensemble methods: RF [33] and AdaBoost. RF is an ensemble approach consisting of many DTs, and it increases the overall result by combine learning models. The number of DTs in RF is set as 50, and the maximum depth of forest is 6. AdaBoost is one of the first boosting algorithms and works by putting more weight on difficult to classify instances. The weak learners in AdaBoost are DTs, and the number of DTs is also 50.

A. Evaluation Metrics

Electricity theft detection can be regarded as a binary classification problem. In this article, malicious samples are denoted as positive class and benign samples as negative class. The performance of proposed methods is evaluated by precision (Pre), FPR, recall, and area under the curve (AUC). The confusion matrix used to evaluate the performance of the electricity theft detection model can be constructed, as shown in Table II.

TABLE III
RESULTS OF DETECTION BY USING THE PROPOSED METHOD FOR SIX ATTACK TYPES

Attack Type	Training ratio 50%				Training ratio 60%				Training ratio 70%				Training ratio 80%			
	Pre (%)	FPR (%)	Recall (%)	AUC (%)	Pre (%)	FPR (%)	Recall (%)	AUC (%)	Pre (%)	FPR (%)	Recall (%)	AUC (%)	Pre (%)	FPR (%)	Recall (%)	AUC (%)
Type 1	92.63	6.174	91.83	98.44	94.09	4.78	93.54	98.93	93.84	4.84	93.26	98.9	**94.56**	**4.55**	**93.82**	**99.08**
Type 2	90.45	10.58	87.77	96.19	**91.48**	10.05	88.62	96.54	91.3	9.73	88.23	**96.84**	91.27	**9.70**	**88.65**	96.08
Type 3	96.69	3.69	94.8	99.41	97.24	2.77	95.98	99.57	**97.64**	**2.27**	**96.8**	99.73	97.52	2.31	96.33	**99.75**
Type 4	92.15	5.03	93.17	98.42	91.69	4.8	92.78	98.63	93.05	3.96	94.3	98.83	**93.13**	3.97	93.9	98.99
Type 5	92.57	1.99	96.26	99.28	93.11	**1.51**	**96.98**	99.38	**93.39**	1.66	96.37	99.51	92.67	1.75	95.59	**99.52**
Type 6	**93.82**	5.80	**93.5**	98.74	93.21	5.97	92.88	**99.87**	93.1	5.62	92.83	98.85	93.11	**5.31**	93.31	98.91
Mixed Type	96.61	**3.49**	93.74	99.52	97.62	4.12	**95.18**	99.44	**97.65**	3.96	94.99	99.52	96.47	3.57	93.78	**99.62**

As shown in Table II, the definition of precision, FPR, and recall can be written as follows:

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{18}$$

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \tag{19}$$

$$\text{FPR} = \text{FP}/(\text{FP} + \text{TN}) \tag{20}$$

where precision is defined as the proportion of positive identifications is actually correct. Recall is defined as the proportion of actual positives was identified correctly. FPR is defined as the percentage of negative samples incorrectly identified as positive samples in the data set.

B. Performance of Proposed Method

In the first experiment, we train our model using both benign and malicious samples. The performance of the proposed method is evaluated in six different attack types separately. Then, all six attack types are mixed to create Type Mix to test the performance. It is worth mentioning that four different training ratios are chosen randomly in this experiment: 50%, 60%, 70%, and 80%. Among the 533 samples, we randomly choose 50% to generated malicious samples. The above procedure is repeated for 500 randomly chosen customers.

Table III presents the values of precision, recall, FPR, and AUC of the proposed method in detecting six attack types and Type Mix in different training rates. From the test results, it can be found that our method has an excellent performance in detecting all attack types except for attack Type 2.

Attack Type 2 is the most difficult type to be detected by our method. This can be explained by the fact that the factor α_t multiplied by meter reading data is constantly changing over time, which allows our model hard to learn. The experiment results indicate that our method shows good performance under different attack types.

C. Performance Comparison

In the second experiment, the proposed method is compared with other conventional methods in terms of precision, recall, FPR, and AUC. We conduct the experiment in the Type Mix and use an 80/20 training/test split on this data set. Table IV shows the detailed values of those evaluation metrics, and the ROC curve is presented in Fig. 3.

TABLE IV
COMPARISONS OF DETECTION ACCURACY USING DIFFERENT METHODS

Methods	Precision (%)	FPR (%)	Recall (%)	AUC (%)
Our method	**97.53**	**3.17**	**93.78**	**99.62**
SVM	93.6	8.13	90.6	97.69
BPNN	82.01	15.76	83.14	91.12
ELM	81.49	18.82	79.9	88.49
DELM	78.4	20.08	78.06	84.43
KNN	87.98	15.65	80.98	93.21
RF	96.53	6.15	93.11	99.08
DT	97.09	6.17	92.42	96.17
LR	82.22	15.32	83.95	91.07
NB	79.7	20.09	78.15	86.77
Adaboost	89.38	15.81	82.39	93.08

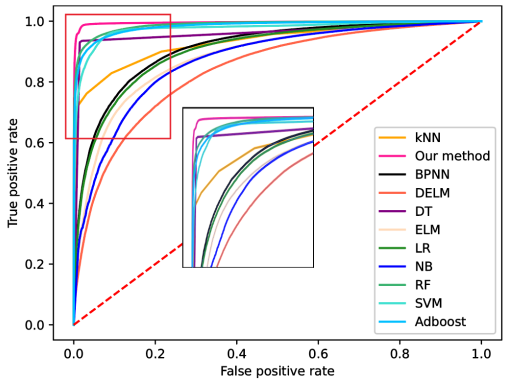


Fig. 3. ROC curves for our method and other AI-based methods.

It indicates that our method achieves the maximum precision value and recall value with 97.53% and 93.78%, respectively, compared with other methods. Our method also shows the lowest FPR among those methods, with 3.17%. Fig. 3 shows the ROC curve of our methods and other methods. As we can see, the proposed method provides a better AUC value compared to other methods.

From this experiment, we find that the results of DELM are the worst among all these methods. The FPR and AUC of DELM are 20.08% and 84.43%, respectively. It indicates that DELM is not a promising method for electricity theft detection. We also find that performance of RF is very close to our method, except that the FPR is 6.15% higher than our

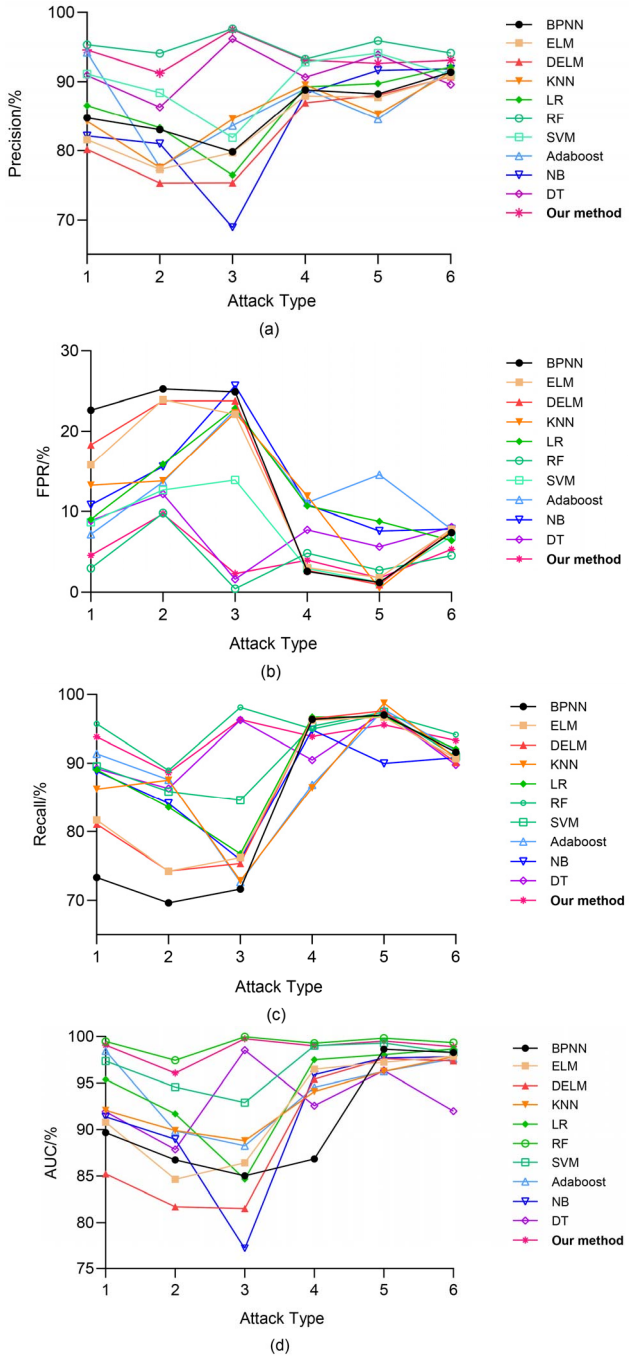


Fig. 4. Performance of the methods in different attack types. (a) Precision values of the methods. (b) FPR values of the methods. (c) Recall values of the methods. (d) AUC values of the methods.

method, with 3.17%. Its other evaluation metrics are only lower than our method by 1%. This can be explained by the fact that RF and our method are used the same model representation and inference, but a different training algorithm.

In the third experiment, we compare the specific performance of all the models in detecting six attack types separately. Fig. 4 shows the precision, FPR, recall, and AUC of all the methods in detecting six attack types separately. The results for the detection of single attack types indicate that different methods have their own advantages under specific attack types.

TABLE V
PERFORMANCE COMPARISON OF OUR METHOD AND SVM IN 10% OF MALICIOUS SAMPLES

Attack Types	Methods	Precision (%)	FPR (%)	Recall (%)	AUC (%)
Attack Type 1	Our method	90.84	7.72	74.86	97.99
	SVM [19]	85.73	2.90	73.657	94.34
Attack Type 2	Our method	89.13	9.51	70.6	94.71
	SVM	79.47	3.39	72.07	92.10
Attack Type 3	Our method	95.32	2.6	83.34	98.83
	SVM	75.66	5.24	50.453	88.38
Attack Type 4	Our method	85.79	12.65	67.7	98.01
	SVM	82.87	2.38	77.417	96.49
Attack Type 5	Our method	87.81	10.73	89.34	98.61
	SVM	92.13	0.88	90.857	98.75
Attack Type 6	Our method	90.52	6.71	77.2	97.86
	SVM	86.90	3.01	69.948	96.45
Mixed Type	Our method	98.93	1.61	84.18	99.11
	SVM	96.76	2.71	74.051	94.93

The results show that proposed methods exhibit good performance in the detection of all six attack types. The ensemble-based methods, especially our method and RF, have quite a high value of precision and low value of FPR in detecting attack Type 1, Type 2, and Type 3, followed by other conventional methods, such as SVM, LR, and DT. It is shown that traditional neural network-based methods have poor performance in the detection of those three attack types, with BPNN the worst off. We also notice that the performance of NN methods is affected by different numbers of hidden layers and neurons during the experiment. Therefore, how to tune the hyperparameters of NN is quite a tough work. However, in the detection of attack Type 4, Type 5, and Type 6, these three kinds of NN-based methods (BPNN, ELM, and DELM) perform good performance, which has relatively high precision and recall. The deep blue line in Fig. 4 shows that NB has the worst performance in detection Type 3 among all methods. In the detection of Type 4, the AUC of our method, RF, and SVM are higher than other methods, and BPNN has poor performance in detecting this attack type. Our method also shows excellent performance in the detection of Type 5 and Type 6. This improvement may due to XGBoost introducing the second-order approximation in the objective function to avoid overfitting.

D. Performance in Imbalanced Data Set

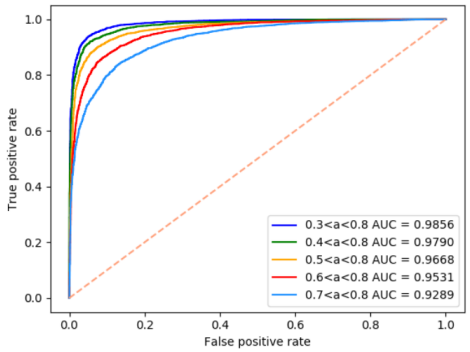
To further analyze the performance of our method, we evaluate the performance of the proposed method when the proportion of malicious samples is 10%, i.e., the performance of the proposed method when training data are imbalanced. We compare the proposed method with the method proposed by Jokar *et al.* [19], and the experiment results are shown in Table V. In addition, we conduct experiments to evaluate the performance when the data set is extremely imbalanced, i.e., the proportion of malicious samples is 1%, and the results are shown in Table VI.

It can be found from Table V that our method shows good performance in terms of precision and FPR, with 95.32% and 2.6% when detecting the attack Type 3, compared

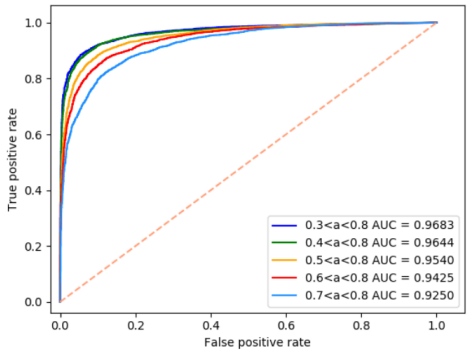
TABLE VI

PERFORMANCE OF THE PROPOSED METHOD IN THE CASE OF 1% MALICIOUS SAMPLES

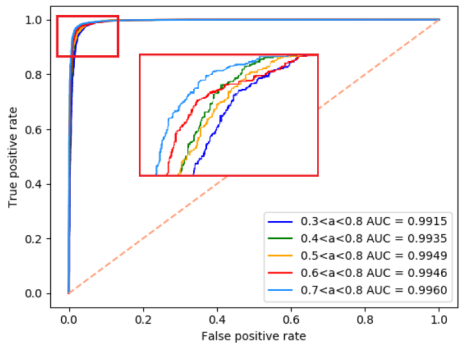
Attack Types	Precision (%)	FPR (%)	Recall (%)	AUC (%)
Attack Type 1	19.80	20.08	16.06	61.22
Attack Type 2	24.95	21.05	19.55	64.26
Attack Type 3	27.00	22.88	20.63	65.11
Attack Type 4	7.74	22.73	6.40	54.67
Attack Type 5	39.70	17.33	36.76	75.32
Attack Type 6	22.30	21.43	19.00	63.98
Mixed Type	90.20	0.81	39.29	69.92



(a)



(b)



(c)

Fig. 5. ROC curves for the proposed method under different attacks. (a) Impact of α for attack Type 1. (b) Impact of α_t for attack Type 2. (c) Impact of α_t for attack Type 4.

to SVM with 75.66% and 5.24%, respectively. However, SVM shows better performance than our method in detecting Type 4. The precision and FPR of SVM are 85.79% and 12.65%, compared with 82.78% and 2.34%, that of our method. A similar situation can be found in the detection of Type 5, and all evaluation metrics of SVM are better than our method. In the detection of Type 1 and Type 4, the precision, recall, and AUC of our method are better than SVM, while the FPR is inferior to SVM. It can be explained by the fact that our method will classify more malicious samples as benign samples in the situation of data are imbalanced. In the detection of mixed-type, our method still shows better performance than SVM. In general, the performance of our method is found to be good in the situation of malicious samples, which is smaller than benign samples and may because XGBoost can be modified to weight error gradients to malicious samples importance proportional during training.

In addition, we evaluate the performance of the proposed method when the proportion of malicious samples is 1%. Table VI shows that the model trained by the extremely imbalanced data sets (1% of malicious samples) shows poor performance. In the real world, the percentage of malicious samples is very small. In such a scenario, it would be unwise to train the classifier directly using these extremely imbalanced data sets. As we can see, it will produce very poor results. Thus, it is necessary to use techniques to preprocess the imbalanced data set, such as resampling, allowing the frequency of each class as far as possible equal.

In order to evaluate the impact of different α's and α_t's under different attack types (i.e., attack Type 1, attack Type 2, and attack Type 4), we choose five pairs of different parameters given as $(0.3, 0.8)$, $(0.4, 0.8)$, $(0.5, 0.8)$, $(0.6, 0.8)$, and $(0.7, 0.8)$ to evaluate the proposed method. Fig. 5 shows that the proposed method has good performance in detecting different attack types with different pairs of α or α_t.

V. CONCLUSION

In this article, we propose an electricity theft detection scheme based on the XGBoost for AMI. In this XGBoost-based electricity theft detector, the gradient boosting builds an ensemble of DTs, continuously adding new trees to correct the errors made by existing model. The XGBoost-based electricity theft detector is trained using both benign samples and malicious samples. Simulations are done by using the Irish Smart Energy Trails with six attack types. Results show that the proposed model outperforms several AI-based models, including the SVM, the backpropagation neural network, ELM, DELM, the kNN algorithm, LR, DT, RF, the NB classifier, and AdaBoost. Results show that the proposed method achieves good performance when the training set is imbalanced. Also, the proposed method provides high performance in different ranges of parameter α.

However, for real-world application, there are some limitations in the proposed method. First, the proposed method only analyzes electricity consumption data alone, which may produce limited results. In addition to meter reading data, other information such as climatic factors (temperature), regional factors, and some electric factors (current and voltage) is worth researching in the future. Second, there may not enough training data in the real world to train the method. To protect the privacy of customers, many electric companies do not provide their customers' data for the purpose of research. In addition, electricity theft detection will inevitably have the problem of imbalanced data, i.e., the number of malicious samples is

greater than the number of benign samples. Unfortunately, there is no one-fit-all solution to handle this problem, and we can only try multiple techniques in the real world. Finally, we make some assumptions in this article, for example, we do not consider the error of smart meters and TLs in the distribution network. These factors must be considered in reality.

REFERENCES

[1] K. Yu, M. Arifuzzaman, Z. Wen, D. Zhang, and T. Sato, "A key management scheme for secure communications of information centric advanced metering infrastructure in smart grid," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 8, pp. 2072–2085, Aug. 2015.

[2] M. Huang, Z. Wei, M. Pau, F. Ponci, and G. Sun, "Interval state estimation for low-voltage distribution systems based on smart meter data," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 9, pp. 3090–3099, Sep. 2019.

[3] J. Zhang, L. Tang, A. Mingotti, L. Peretto, and H. Wen, "Analysis of white noise on power frequency estimation by DFT-based frequency shifting and filtering algorithm," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 7, pp. 4125–4133, Jul. 2020.

[4] L. N. Group. *World Loses 89.3 Billion to Electricity Theft Annually, 58.7 Billion in Emerging Markets*. Accessed: Dec. 9, 2014. [Online]. Available: https://eepower.com/news/world-loses-89-3-billion-annually-to-electricity-theft/

[5] *FBI: Smart Meter Hacks Likely to Spread*. Accessed: Apr. 12, 2012. [Online]. Available: https://krebsonsecurity.com/2012/04/fbi-smart-meter-hacks-likely-to-spread/

[6] B. H. P. Smart. *Smart Meters Help Reduce Electricity Theft, Increase Safety*. Accessed: May 3, 2011. [Online]. Available: https://www.nationalgeographic.com/news/energy/2011/09/110913-smart-meters-for-electricity-theft/

[7] R. Katakey. *India Fights to Keep the Lights On*. Accessed: Jun. 5, 2014. [Online]. Available: https://www.bloomberg.com/news/articles/2014-06-05/india-fights-electricity-theft-as-modi-pledges-energy-upgrade

[8] T. Hu, Q. Guo, X. Shen, H. Sun, R. Wu, and H. Xi, "Utilizing unlabeled data to detect electricity fraud in AMI: A semisupervised deep learning approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3287–3299, Nov. 2019.

[9] Z. Shuai, J. Zhang, L. Tang, Z. Teng, and H. Wen, "Frequency shifting and filtering algorithm for power system harmonic estimation," *IEEE Trans. Ind. Informat.*, vol. 15, no. 3, pp. 1554–1565, Mar. 2019.

[10] R. Jiang, R. Lu, Y. Wang, J. Luo, C. Shen, and X. Shen, "Energy-theft detection issues for advanced metering infrastructure in smart grid," *Tsinghua Sci. Technol.*, vol. 19, no. 2, pp. 105–120, Apr. 2014.

[11] H. O. Henriques *et al.*, "Development of adapted ammeter for fraud detection in low-voltage installations," *Measurement*, vol. 56, pp. 1–7, Oct. 2014.

[12] S.-C. Huang, Y.-L. Lo, and C.-N. Lu, "Non-technical loss detection using state estimation and analysis of variance," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2959–2966, Aug. 2013.

[13] C.-H. Lo and N. Ansari, "CONSUMER: A novel hybrid intrusion detection system for distribution networks in smart grid," *IEEE Trans. Emerg. Topics Comput.*, vol. 1, no. 1, pp. 33–44, Jun. 2013.

[14] Z. Xiao, Y. Xiao, and D. H.-C. Du, "Non-repudiation in neighborhood area networks for smart grid," *IEEE Commun. Mag.*, vol. 51, no. 1, pp. 18–26, Jan. 2013.

[15] H. O. Henriques, R. L. S. Correa, M. Z. Fortes, B. S. M. C. Borba, and V. H. Ferreira, "Monitoring technical losses to improve non-technical losses estimation and detection in LV distribution systems," *Measurement*, vol. 161, Sep. 2020, Art. no. 107840.

[16] A. A. Cardenas, S. Amin, G. Schwartz, R. Dong, and S. Sastry, "A game theory model for electricity theft detection and privacy-aware control in AMI systems," in *Proc. 15th Annu. Allerton Conf.*, Allerton House, UIUC, IL, USA, Oct. 2012, pp. 1830–1837.

[17] Z. Zhou, J. Bai, M. Dong, K. Ota, and S. Zhou, "Game-theoretical energy management design for smart cyber-physical power systems," *Cyber-Physical Syst.*, vol. 1, no. 1, pp. 24–45, Jan. 2015.

[18] S. S. S. R. Depuru, L. Wang, V. Devabhaktuni, and P. Nelapati, "A hybrid neural network model and encoding technique for enhanced classification of energy consumption data," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Jul. 2011, pp. 1–8.

[19] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Electricity theft detection in AMI using Customers' consumption patterns," *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 216–226, Jan. 2016.

[20] Z. Zheng, Y. Yang, X. Niu, H.-N. Dai, and Y. Zhou, "Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids," *IEEE Trans. Ind. Informat.*, vol. 14, no. 4, pp. 1606–1615, Apr. 2018.

[21] R. Punmiya and S. Choe, "Energy theft detection using gradient boosting theft detector with feature engineering-based preprocessing," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 2326–2329, Mar. 2019.

[22] E. W. S. Angelos, O. R. Saavedra, O. A. C. Cortés, and A. N. de Souza, "Detection and identification of abnormalities in customer consumptions in power distribution systems," *IEEE Trans. Power Del.*, vol. 26, no. 4, pp. 2436–2442, Oct. 2011.

[23] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.

[24] Z. Yan and H. Wen, "Electricity theft detection base on extreme gradient boosting in AMI," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC)*, May 2020, pp. 1–6.

[25] *Irish Smart Energy Trial*. Accessed: Sep. 2012. [Online]. Available: http://www.ucd.ie/issda/data/commissionforenergyregulationcer/

[26] A. Abdallah and X. Shen, "Lightweight security and privacy preserving scheme for smart grid customer-side networks," *IEEE Trans. Smart Grid*, vol. 8, no. 3, pp. 1064–1074, May 2017.

[27] S. McLaughlin, B. Holbert, A. Fawaz, R. Berthier, and S. Zonouz, "A multi-sensor energy theft detection framework for advanced metering infrastructures," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 7, pp. 1319–1330, Jul. 2013.

[28] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Belmont, CA, USA: Whadsworth International Group, 1984.

[29] A. H. Nizar, Z. Y. Dong, and Y. Wang, "Power utility nontechnical loss analysis with extreme learning machine method," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 946–955, Aug. 2008.

[30] C. Cody, V. Ford, and A. Siraj, "Decision tree learning for fraud detection in consumer energy consumption," in *Proc. IEEE 14th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2015, pp. 1175–1179.

[31] S. Aziz, S. Z. Hassan Naqvi, M. U. Khan, and T. Aslam, "Electricity theft detection using empirical mode decomposition and K-Nearest neighbors," in *Proc. Int. Conf. Emerg. Trends Smart Technol. (ICETST)*, Mar. 2020, pp. 1–5.

[32] P. Glauner, J. A. Meira, L. Dolberg, R. State, F. Bettinger, and Y. Rangoni, "Neighborhood features help detecting non-technical losses in big data sets," in *Proc. 3rd IEEE/ACM Int. Conf. Big Data Comput., Appl. Technol.*, Dec. 2016, pp. 253–261.

[33] I. Monedero, F. Biscarri, C. León, J. I. Guerrero, J. Biscarri, and R. Millán, "Detection of frauds and other non-technical losses in a power utility using pearson coefficient, Bayesian networks and decision trees," *Int. J. Electr. Power Energy Syst.*, vol. 34, no. 1, pp. 90–98, Jan. 2012.

Zhongzong Yan (Student Member, IEEE) was born in Guangxi, China, in 1995. He received the B.Sc. degree in electrical engineering from Guangxi University, Nanning, China, in 2018. He is currently pursuing the M.Sc. degree with the College of Electrical and Information Engineering, Hunan University, Changsha, Hunan, China.

His research interests include machine learning and its application in smart grids.

He Wen (Senior Member, IEEE) was born in Hunan, China, in 1982. He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Hunan University, Hunan, in 2004, 2007, and 2009, respectively.

He is currently a Full Professor with the College of Electrical and Information Engineering, Hunan University, where he is also the Deputy Director of the Hunan Province Key Laboratory of Intelligent Electrical Measurement and Application Technology. His current research interests include electrical contact reliability, wireless communications, power system harmonic measurement and analysis, power quality, and digital signal processing.

Dr. Wen is also an Associate Editor of the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, an Academic Editor of the *Journal of Sensors*, and a member of the Editorial Board of *Fluctuation and Noise Letters* and *China Test and Measurement*.