Wide & Deep Convolutional Neural Networks for Electricity-Theft Detection to Secure Smart Grids

Zibin Zheng, *Senior Member, IEEE,* Yatao Yang, Xiangdong Niu, Hong-Ning Dai, *Senior Member, IEEE*, Yuren Zhou

***Abstract*—Electricity theft can be harmful to power grid suppliers and cause economic losses. Integrating information flows with energy flows, smart grids can help to solve the problem of electricity theft owning to the availability of massive data generated from smart grids. The data analysis on the data of smart grids is helpful in detecting electricity theft because of the abnormal electricity consumption pattern of energy thieves. However, the existing methods have poor detection accuracy of electricity-theft since most of them were conducted on one dimensional (1-D) electricity consumption data and failed to capture the periodicity of electricity consumption. In this paper, we originally propose a novel electricity-theft detection method based on Wide & Deep Convolutional Neural Networks (CNN) model to address the above concerns. In particular, Wide & Deep CNN model consists of two components: the Wide component and the Deep CNN component. The Deep CNN component can accurately identify the non-periodicity of electricity-theft and the periodicity of normal electricity usage based on two dimensional (2-D) electricity consumption data. Meanwhile, the Wide component can capture the global features of 1-D electricity consumption data. As a result, Wide & Deep CNN model can achieve the excellent performance in electricity-theft detection. Extensive experiments based on realistic dataset show that Wide & Deep CNN model outperforms other existing methods.**

***Index Terms*—Electricity Theft Detection, Smart Grids, Convolutional Neural Networks, Machine Learning, Deep Learning**

I. Introduction

Electricity has become a necessity in our life. Losses often occur during electricity generation, transmission and distribution. The electricity losses can be generally categorized into technical losses (TLs) and Non-technical losses (NTLs) [1]. One of the primary NTLs is *electricity theft*. This misbehavior usually includes bypassing the electricity meter, tampering the meter reading, or hacking the meter [2]. Electricity theft can result in the surging electricity, the heavy load of electrical systems, the huge revenue loss of power company and the dangers to public safety (such as fires and electric shocks). For example, it is reported in [3] that the losses due to electricity theft approximate 100 million Canadian dollars every year; this lost electricity can even supply 77,000 homes for a year.

There is a substantial body of studies on detecting electricity theft. Conventional electricity-theft detection methods include: humanly checking problematic meter installation or mis-configuration, comparing the abnormal meter readings with the normal ones and examining the bypassed power transmission line etc. However, these methods are extremely time-consuming, expensive and inefficient.

The appearance of *smart grids* brings opportunities in solving electricity theft. Smart grids essentially consist of traditional power grids, communications networks connecting intelligent devices (such as smart meters and sensors) in grids and computing facilities to sense and control grids [4]. Both energy flows and information flows in smart grids connect users and utility companies together. In this manner, smart meters or sensors can collect data such as electricity usage, status information of grids, electricity price and financial information [5]. The data of smart grids is helpful for us to design demand response management (DRM) schemes [6], forecast the electricity price [7] and schedule the electricity in more profitable way [8], [9]. In addition, some recent works such as [2], [10], [11], [12], [13] show that data analysis on smart grids can help to detect electricity theft. However, most of these approaches have the following limitations: 1) many of them require specific devices [1]; 2) most of them are based on artificial feature extraction according to domain knowledge (requiring manual interventions); 3) many methods (such as support vector machine and linear regression) have low electricity-theft detection accuracy.

Therefore, in this paper, we aim to design a novel electricity-theft detection method to address the above concerns. In particular, we originally propose a Wide & Deep Convolutional Neural Networks (CNN) model to learn the electricity consumption data and identify the electricity thieves. Our Wide & Deep CNN model consists of a Wide component with a fully-connected layer of neural networks and a Deep CNN component with multiple convolutional layers, a pooling layer and a fully-connected layer. Essentially, the Wide component can learn the global knowledge while the Deep CNN component can capture the periodicity of electricity consumption data. This model integrates the benefits of the Wide component and the Deep CNN component consequently resulting in good performance in electricity-theft detection.

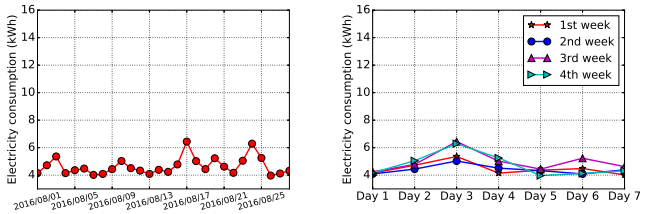
The primary research contributions of this paper can be summarized as follows.

- We originally propose a Wide & Deep CNN model to analyze electricity theft in smart grids. *To the best of our*

Manuscript received August 31, 2017. The work described in this paper was supported by the National Key Research and Development Program (2016YFB1000101), the National Natural Science Foundation of China (61722214, 61472338), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (2016ZT06D211), and the Pearl River S & T Nova Program of Guangzhou (201710010046). Paper No. TII-17-2030. (*Corresponding author: H.-N. Dai)*

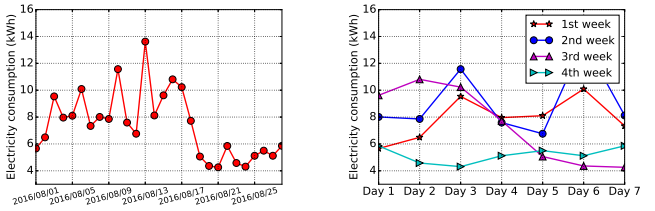
Z. Zheng, Y. Yang, X. Niu and Y. Zhou is with School of Data and Computer Science, Sun Yat-sen University, China (email:zhzibin@mail.sysu.edu.cn; yangyt9@mail2.sysu.edu.cn; niuxd@mail2.sysu.edu.cn; zhouyuren@mail.sysu.edu.cn).

H.-N. Dai is with Faculty of Information Technology, Macau University of Science and Technology, Macau SAR (email:hndai@ieee.org).



(a) Electricity consumption (kWh) by date (b) Electricity consumption (kWh) by week

Fig. 1. An example of electricity consumption of normal usage



(a) Electricity consumption (kWh) by date (b) Electricity consumption (kWh) by week

Fig. 2. An example of electricity consumption of electricity theft

knowledge, it is the first study to propose the Wide & Deep CNN model and apply it to analyze electricity theft in smart grids.

- Our model has numerous merits: 1) *memorization* of the global knowledge brought by the Wide component, 2) *generalization* of the new knowledge brought by the Deep CNN model, 3) *accuracy* in electricity-theft detection.
- We have conducted extensive experiments on massive realistic electricity consumption dataset. Experimental results show that our Wide & Deep CNN model outperforms than other existing approaches.

The remainder of the paper is organized as follows. Section II presents an overview on related literature. We present the problem analysis in Section III. Section IV presents the Wide & Deep CNN model. We then give the experimental results in Section V. Finally, we conclude the paper in Section VI.

II. Related work

In this section, we first present a survey on electricity-theft detection in Section II-A and an overview on anomaly detection in Section II-B.

A. Electricity-theft Detection

We roughly categorize the studies on electricity-theft detection into two types: hardware-based solutions and data-driven solutions. In particular, hardware-based solutions concentrate on designing specific metering devices and infrastructures so that electricity theft can be easily detected. Typical electricity-theft detection equipments include smart meters with anti-tampering sensors, radio-frequency identification (RFID) tags and sensors [14], [15], [16]. The main limitations of hardware-based solutions include 1) the cost of deploying smart metering devices, 2) the vulnerability of hardware devices (e.g., failure due to severe weather condition), 3) the difficulty in maintaining devices (e.g., replacing batteries of devices).

Data driven electricity-theft detection has drawn considerable attentions recently. For example, the work in [2] is based on the data fusion from sensors and advanced metering infrastructure (AMI). Many recent studies [17], [18], [19] are based on support vector-machines (SVM). The main idea of SVM methods is to classify the normal users and the electricity thieves. In addition to SVM, artificial neural networks can also be used to electricity-theft detection [10], [11]. However, most of these studies are less accurate in electricity-theft detection and require artificial feature extraction according to domain knowledge.

B. Anomaly Detection in Smart Grids

Anomaly detection in smart grids represents a substantial body of works related to data driven electricity-theft detection. In particular, anomaly detection (a.k.a. outlier detection) is the procedure of detecting abnormal patterns that do not conform the expected behavior [20]. Anomaly detection has been widely used in many research areas, such as intrusion detection [21], fraud detection [22] and industrial control systems [23].

Recently, anomaly detection has received extensive attention from the smart grid community since it can help in improving operational safety, enhancing the control reliability and detecting faults in smart metering infrastructure [24], [25], [26]. The typical approaches used in anomaly detection in smart grids mainly include SVM (Support Vector Machine), clustering and classification [27]. Besides, Decision Tree and Rough Sets can also be used in fraud detection in power systems [28]. Moreover, [29] presents a rule-based model to detect the NTLs.

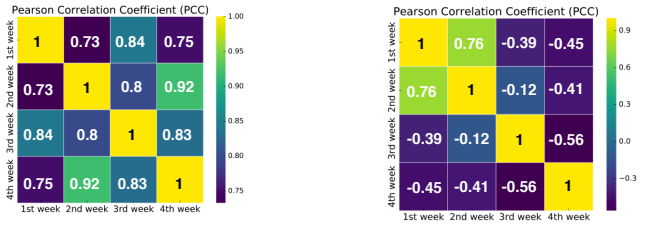
However, most of related studies in either electricity-theft detection or anomaly detection are based on the analysis on 1-D electricity consumption data and fail to capture the periodicity of electricity consumption. Therefore, it is the purpose of this study to propose a novel analytical model to overcome the limitations of the above existing works.

III. Problem Analysis

Electricity theft is a criminal behavior of stealing electrical power from power grids. This malicious behavior can be done by bypassing the electricity meter, tampering the meter reading, or hacking the meter. Since electricity theft can result in the abnormal patterns of electricity consumption, the data-driven electricity-theft detection approaches have received extensive attention recently due to the availability of smart-meter readings and electricity consumption data from smart grids. We next illustrate the abnormality of electricity consumption data of energy thieves, which can be potentially captured by machine learning tools.

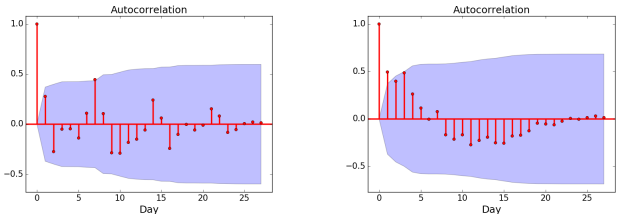
We conduct a preliminary analysis on electricity consumption data. This dataset released by State Grid Corporation of China (SGCC)[1] contains the electricity consumption data of 42,372 electricity customers within 1,035 days (details about the dataset will be given in Section V-A). In particular, Fig. 1 (a) shows an example of electricity consumption of normal

[1]State Grid Corporation of China http://www.sgcc.com.cn/



(a) PCC values of normal customers (b) PCC values of energy thieves

Fig. 3. Pearson Correlation Coefficient (PCC) of electricity consumption



(a) ACF values of normal customers (b) ACF values of energy thieves

Fig. 4. Autocorrelation function (ACF) of electricity consumption by week

usage by a customer in a month (i.e., August 2016). We observe that there is a fluctuation on the electricity consumption data day by day. It is hard to capture the key characteristics of electricity thieves and normal customers from this 1-D data. However, we can observe that there is a periodicity on the electricity consumption of this customer if we plot the data in 2-D manner by week as shown in Fig. 1 (b), in which the electricity consumption reaches the peak on day 3 every week while it often reaches the bottom on day 5 every week (the exception is on the 2nd week, when there is the lowest consumption on day 6). In fact, we can have similar findings for the whole dataset (i.e., electricity consumption data with 1,035 days). Without too many repetitions, we only show an excerpt data from the whole dataset. We can observe that there is a *periodicity* for most of normal customers if we align the electricity consumption data of all the 35 months together.

In contrast, Fig. 2 shows an example of electricity consumption of an electricity theft in a month. Similar to Fig. 1, we also plot the electricity consumption by date (as shown in Fig. 2 (a)) and the electricity consumption by week (as shown in Fig. 2 (b)). As shown in Fig. 2, we observe that the electricity consumption in the first two weeks (i.e., the 1st week and the 2nd week) fluctuates periodically. For example, the electricity consumption reaches the peak on day 3 and on day 6 every week. However, there is a distinct loss of the electricity consumption from the third week and the electricity consumption has remained at the low level after that.

To better analyze the periodicity of normal customers and non-periodicity of energy thieves, we conduct a correlation analysis on the electricity consumption data. Fig. 3 shows Pearson Correlation Coefficient (PCC) of electricity consumption of both normal customers and energy thieves by week. In particular, Fig. 3 (a) shows PCC values of normal customers while Fig. 3 (b) shows those of energy thieves. We can find from Fig. 3 (a) that there is a strong correlation of electricity consumption data of normal customers, i.e., most of PCC values are greater than 0.8 (a larger PCC value close to 1 means the stronger correlation [30]). However, we cannot observe the correlation of electricity consumption data of energy thieves as shown in Fig. 3 (b) (i.e., most of PCC values are less than 0.7 and many of them are even negative). We further plot autocorrelation function (ACF) of the electricity consumption data of both normal customers and energy thieves by day in Fig. 4. Fig. 4 shows ACF of the electricity consumption data of normal customers by day in contrast to that of energy thieves. It is observed from Fig. 4 that the electricity consumption patterns of normal customers have obvious periodicity, i.e., the similar pattern longs for about 7 days (see Fig. 4 (a)) while there is no obvious periodicity of electricity consumption of energy thieves (see Fig. 4 (b)).

After statistically analyzing the electricity consumption data of both normal customers and energy thieves, we can find that *the electricity consumption data of energy thieves is usually less periodic or non-periodic, compared with that of normal customers*. We believe that this observation can also be confirmed by other electricity consumption datasets from different countries and regions as implied by recent work [31]. This observation has motivated us to investigate the periodicity of the electricity consumption and identify the abnormal electricity usage.

However, it is challenging to analyze the periodicity of the electricity consumption data due to the following reasons: 1) the electricity consumption data is often erroneous and noisy; 2) it is difficult to analyze the periodicity of the electricity consumption data since it is 1-D time series data with massive size; 3) many conventional data analysis approaches such as Support Vector Machine (SVM) and Artificial Neural Network (ANN) cannot be directly applied to the electricity consumption data due to the computation complexity and the limited generalization capability [32], [33], [31], [7]. In order to address the above challenges, we propose Wide & Deep Convolutional Neural Networks framework (CNN), which will be described in detail in Section IV.

IV. Our Approach

A. Data Preprocessing

Electricity consumption data often contains missing or erroneous values; this is mainly caused by various reasons such as the failure of smart meters, the unreliable transmission of measurement data, the unscheduled system maintenance and storage issues [34]. In this paper, we exploit the interpolation method to recover the missing values according to the following equation,

$$f(x_i) = \begin{cases} \frac{x_{i-1}+x_{i+1}}{2} & x_i \in \mathsf{NaN}, x_{i-1}, x_{i+1} \notin \mathsf{NaN} \\ 0 & x_i \in \mathsf{NaN}, x_{i-1} \ or \ x_{i+1} \in \mathsf{NaN} \\ x_i & x_i \notin \mathsf{NaN}, \end{cases} \quad (1)$$

where x_i stands for the value in the electricity consumption data over a period (e.g., a day). If x_i is a null or a non-numeric character, we represent it as NaN (NaN is a set).

Moreover, we have also found that there are erroneous values (i.e., outliers) in the electricity consumption data. In

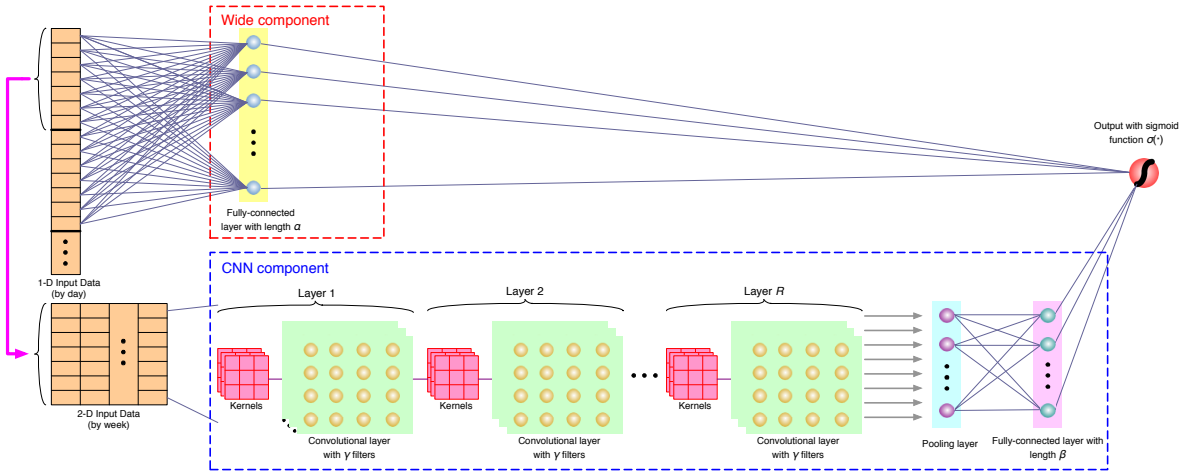


Fig. 5. Wide & Deep Convolutional Neural Networks (CNN) framework

particular, we restore the value by the following equation according to "Three-sigma rule of thumb" [35],

$$f(x_i) = \begin{cases} \text{avg}(\mathbf{x}) + 2 \cdot \text{std}(\mathbf{x}) & \text{if } x_i > \text{avg}(\mathbf{x}) + 2 \cdot \text{std}(\mathbf{x}), \\ x_i & \text{otherwise}, \end{cases} \tag{2}$$

where $\mathbf{x}$ is a vector that is composed of x_i day by day, $\text{avg}(\mathbf{x})$ is the average value of $\mathbf{x}$ and $\text{std}(\mathbf{x})$ is the standard deviation of $\mathbf{x}$. Note that we only consider the positive deviation in Eq. (2). This is because the electricity consumption of each user is always greater than 0 after analyzing the electricity consumption data of 1,035 days. In summary, this method can effectively mitigate the outliers.

After dealing with the missing values and the outliers, we need to normalize the electricity consumption data because the neural network is sensitive to the diverse data. In particular, we choose the MAX-MIN scaling method to normalize the data according to the following equation,

$$f(x_i) = \frac{x_i - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})}, \tag{3}$$

where $\min(\mathbf{x})$ is the minimum value in $\mathbf{x}$, and $\max(\mathbf{x})$ is the maximum value in $\mathbf{x}$.

B. Wide & Deep CNN Framework

As shown in Fig. 5, the Wide & Deep CNN framework mainly consists of two major components: the Wide component and the Deep CNN component. We then explain them in detail as follows.

1) Wide component: As shown in Fig. 5, the Wide component (enclosed in the red dash box) is a fully-connected layer of neural networks and it learns the global knowledge from the 1-D electricity consumption data. According to the preliminary analysis in Section III, the electricity consumption of customers fluctuates from time to time while the normal electricity usage reveals the periodicity and the electricity consumption of energy thieves is less periodic or non-periodic. The electricity consumption of one customer is essentially one dimensional (1-D) time series data. Motivated by the previous study [36], we choose the Wide component to learn the frequent co-occurrence of features by *memorizing* the 1-D time series data.

Every neuron in the fully-connected layer calculates its own score by using the 1-D electricity consumption data according to the following equation,

$$y_j := \sum_{i=1}^{n} \mathbf{w}_{i,j} x_i + b_1, \tag{4}$$

where y_j is the output of the fully-connected layer in the j-th neuron, n is the length of 1-D input data ($\mathbf{x}$), $\mathbf{w}_{i,j}$ stands for the neuron weight between i-th input value and j-th neuron and b_1 is the bias. After the calculation, it will send this value to the connected units in the higher layer through an activation function to determine how much it contributes to the next step prediction. The activation function is given as follows,

$$u_j := f(y_j) = \max(0, y_j), \tag{5}$$

where u_j is the output after activation calculation and $f(\cdot)$ stands for the activation function. In this paper, we use Rectified Linear Unit (ReLU) as the activation function, which will only activate the positive value. This function can effectively prevent the overfitting [37]. This procedure is called the forward pass. The back-propagation works in an opposite direction. During the back-propagation, every unit computes its weights according to the loss value sent from higher layer.

2) Deep CNN component: Our preliminary analytical results (refer to Section III) reveal the periodicity of the normal electricity usage and the non-periodicity of the electricity theft. However, it is difficult to identify the periodicity or the non-periodicity of electricity usage from the 1-D electricity consumption data since the electricity consumption in every day fluctuates in a relatively independent way. Nevertheless, our preliminary results also imply that we can easily identify the abnormal electricity usage if we analyze the electricity consumption by aligning the consumption data of several weeks together. The previous work in [36] also indicates that the deep learning can help to derive new features (i.e., *generalization*). Motivated by this observation, we design a Deep CNN component to process the electricity consumption data in 2 dimension (2-D) manner. In particular, we transform

the 1-D electricity consumption data into 2-D data according to weeks. It is worth mentioning that we can transform 1-D data into 2-D data according to different number of days (e.g., 10 days or 15 days). However, we have found that the weekly transformation in practice has the best performance compared with other types of transformations (confirming our previous observation in Section III).

We next let the Deep CNN component be trained on the 2-D electricity consumption data. Fig. 5 shows that Deep CNN component (enclosed in the blue dash box) consists of multiple convolutional layers, a pooling layer and a fully-connected layer. We then explain them in details as follows.

Multiple convolutional layers. One of the limitations of regular neural networks is the poor scalability due to the full-connectivity of neurons. CNN overcomes the disadvantages of regular neural networks by connecting each neuron to its neighboring neurons (not all the neurons). The local region consisting of a small set of neurons is also called as the *receptive field* [38]. Then, a filter (i.e., a vector) with the same size of the receptive field will be used to conduct the convolution operation with the input 2-D data. One convolution layer essentially consists of a number of filters working independently. When the input 2-D data is passing through the filters, convolution operations are conducted. We assume that there are R CNN layers connecting adjacently as shown in Fig. 5. In this manner, we finally obtain a 2-D feature map. In this paper, we choose γ filters; the number of filters γ is adjustable in the experiments. Moreover, we also design unique kernels to fulfill the periodicity of electricity consumption data. We next describe the technical details as follows.

We use the 2-D convolution layer to extract the periodic features from the input 2-D electricity consumption data. We denote the electricity consumption of a customer of the p-th week by vector $\mathbf{v}_p \in \mathbf{R}^d$, where d is 7 in this paper. Note that we choose $d = 7$ because a week has 7 days; this setting is implied by our observation in Section III. We then represent the concatenated electricity consumption of m weeks (denoted by $\mathbf{v}_{1:m}$) as follows,

$$\mathbf{v}_{1:m} := \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ ... \\ \mathbf{v}_m \end{bmatrix}, \tag{6}$$

where $m = \lceil \frac{n}{7} \rceil$. In general, let $\mathbf{v}_{p:p+k}$ refer to the concatenation of weekly electricity consumption $\mathbf{v}_p, \mathbf{v}_{p+1}, ..., \mathbf{v}_{p+k}$.

Traditionally, a convolution operation involves a filter $\hat{\mathbf{w}}$, which is applied to a window of size 3×3 to produce a new feature. This method is effective in image processing and recognition. However, the 2-D electricity consumption data is different from the image data. Therefore, we design the unique kernels to process the 2-D electricity consumption data. In particular, we consider a feature $\mathbf{c}_{p+1,q+1}$, which can be generated from a window of matrix $\mathbf{v}_{p:p+2,q:q+2}$ by the following equation,

$$\mathbf{c}_{p+1,q+1} := f(\hat{\mathbf{w}}(g_1(\mathbf{v}_{p:p+2,p:p+2}) + g_2(\mathbf{v}_{p:p+2,p:p+2})) + b_2), \tag{7}$$

where $b_2 \in R$ is a bias term and $f(\cdot)$ is a non-linear function such as the hyperbolic tangent [38]. It is worth noting that we design kernel functions $g_1(\cdot)$ and $g_2(\cdot)$ dedicated for the 2-D electricity consumption data. In particular, $g_1(\cdot)$ is defined as follows,

$$\begin{aligned} g_1(\cdot) &:= g_1(\mathbf{v}_{p:p+2,p:p+2}) \\ &:= \begin{bmatrix} 2\mathbf{v}_{p,p:p+2} - \mathbf{v}_{p+1,p:p+2} - \mathbf{v}_{p+2,p:p+2} \\ 2\mathbf{v}_{p+1,p:p+2} - \mathbf{v}_{p,p:p+2} - \mathbf{v}_{p+2,p:p+2} \\ 2\mathbf{v}_{p+2,p:p+2} - \mathbf{v}_{p,p:p+2} - \mathbf{v}_{p+1,p:p+2} \end{bmatrix}. \end{aligned} \tag{8}$$

Essentially, $g_1(\cdot)$ transforms the current row values by subtracting the other two rows; *it captures fluctuations and trends in different periods*.

We define $g_2(\cdot)$ as follows,

$$g_2(\cdot) := g_1(\mathbf{v}^T_{p:p+2,p:p+2})^T. \tag{9}$$

Specifically, $g_2(\cdot)$ transforms current column values by subtracting the other two columns; *it essentially captures fluctuations and trends in the same period*.

We then apply this filter to data blocks denoted by $\mathbf{v}_{1:3,1:3}, \mathbf{v}_{1:3,2:4}, ..., \mathbf{v}_{m-2:m,m-2:m}$, respectively to produce a feature map as given as follows,

$$\mathbf{c} := \begin{bmatrix} \mathbf{c}_{1,1} & \cdots & \mathbf{c}_{1,7} \\ \vdots & \ddots & \vdots \\ \mathbf{c}_{m,1} & \cdots & \mathbf{c}_{m,7} \end{bmatrix} \tag{10}$$

where $\mathbf{c}$ is the generated feature map having the same size as the raw matrix because the zero-padding of 1 is used.

Pooling layer. The pooling layer has been typically used in CNN to reduce the number of parameters (e.g., training weights and filters) and the redundant features. Besides, a pool layer can also be used to control the convergence of neural networks (e.g., avoid overfitting). One of the most typical pooling operations is the *max* pooling [39]. In this way, the pooling layer will choose the maximum value of the field covered by the pooling filter. We also use the max pooling operation in this paper. In particular, we define the max pooling operation as $\hat{\mathbf{c}} = \max(\mathbf{c})$, which takes the generated feature map (defined in Eq. (10)) as the inputs. Essentially, there are two directed passing procedures in CNN: 1) during the forward passing procedure, the pooling layer will choose the maximum value of the field covered by the pooling filter; 2) during the back-propagation procedure, the pooling layer will route the gradients (weights) to the preceding filter with the highest value.

Fully-connected layer. The fully-connected layer in the Deep CNN component is similar to that in the Wide component while they are different in size (i.e., the size of the Wide component is α and that of CNN is β). The fully-connected layer in the Deep CNN component is used to obtain the principal features, which can be calculated in a similar way to the fully-connected layer in the Wide component like Eq. (4) and Eq. (5).

Finally, the Wide component and the Deep CNN component are combined using a weighted sum of their output as hidden features; these features are then fed to one *logistic loss* function for the joint training and prediction [36]. In the joint training and prediction procedure, we consider the sum of the weights of both the Wide component and the Deep CNN

TABLE I
META DATA INFORMATION

Description	Value
Time window for electricity consumption	2014/01/01 - 2016/10/31
# of the customers (total)	42,372
# of normal electricity customers	38,757
# of electricity thieves	3,615

component together and optimize the generated parameters at the same time. Motivated by the similar idea of the Wide & Deep learning model [36], we have achieved the joint training and prediction procedure by back-propagation. In particular, the prediction of the model is finally given as follows,

$$P(\mathbf{Y}=1|\mathbf{x}) := \delta(\mathbf{W}[\mathbf{x}_{\text{Wide}}, \mathbf{x}_{\text{CNN}}] + \mathbf{b}) \tag{11}$$

where $\mathbf{Y}$ is the binary class label, $\delta(\cdot)$ is the sigmoid function, $\mathbf{x}_{\text{Wide}}$ and $\mathbf{x}_{\text{CNN}}$ represent the features of the Wide component and those of the CNN component, respectively, $\mathbf{W}$ is the joint weights of the Wide component and the CNN component, and $\mathbf{b}$ is the bias term.

V. EXPERIMENTAL RESULTS

A. Experiment Settings

1) Raw electricity consumption data: We conduct the experiments on a realistic electricity consumption dataset released by State Grid Corporation of China (SGCC). Table I presents the meta data information of this dataset, where # means "the number of". This dataset contains the electricity consumption data of 42,372 electricity customers within 1,035 days (from Jan. 1, 2014 to Oct. 31, 2016). It is worth mentioning that the dataset contains some erroneous data and missing values. Therefore, we exploit the data preprocessing method as described in Section IV-A to address this issue.

2) Ground truth: Along with the released dataset, SGCC also explicitly indicated that the dataset contains 3,615 electricity thieves (as shown in Table I), which occupy nearly 9% of all the customers; this implies that the electricity theft in China is quite serious. We use the the given electricity thieves as the ground truth to evaluate the performance of the proposed scheme and other related schemes.

3) Performance metrics: In this paper, we conduct the experiments by considering two performance metrics: area under curve (AUC) [40] and mean average precision (MAP) [41]. We briefly introduce them as follows.

AUC is often used to evaluate the classification/rank accuracy. The AUC value is equivalent to the probability that a randomly chosen positive sample ranks higher than a randomly chosen negative sample. The equivalent formula for AUC calculation is given as follows,

$$\text{AUC} = \frac{\sum_{i \in \text{positiveClass}} \text{Rank}_i - \frac{M(1+M)}{2}}{M \times N}, \tag{12}$$

where Rank_i represents the rank value of sample i, M is the number of positive samples and N is the number of negative samples. It is worth mentioning that samples are sorted in the ascending order according to the prediction of positive samples for scoring.

MAP is often used to judge the quality of information retrieval. In this paper, we use this metric to evaluate the accuracy of our model. Before using MAP to evaluate, the label of test set is sorted according to the prediction score. We then choose the top N labels to evaluate the performance.

In order to calculate MAP, we first define precision at k (denoted by P@k) as follows,

$$\text{P@}k = \frac{Y_k}{k}, \tag{13}$$

where Y_k denotes the number of correctly-predicted electricity thieves before the location k.

We then denote the mean of all P@k situations by MAP@N (with top N labels), which is given as follows,

$$\text{MAP@}N = \frac{\sum_{i=1}^{r} \text{P@}k_i}{r}, \tag{14}$$

where r is the number of electricity thieves in top N labels and k_i ($i = 1...r$) is the position of the electricity theft.

TABLE II
EXPERIMENT PARAMETERS FOR BASELINE METHODS

Method	Features	Parameters
TSR	Raw (1-D)	$\theta = 1, 2, 3, 4$
LR	Raw (1-D)	Penalty: L2 Inverse of regularization strength: 1.0
RF	Raw (1-D)	# of trees in forest: 200 Function to measure the quality of a split: gini
SVM	Raw (1-D)	Penalty parameter of the error term: 1.0 kernel: radial basis function (RBF)
Wide	Raw (1-D)	same as Wide component of our approach
CNN	Raw (2-D)	same as CNN component of our approach

B. Performance Comparison

In this section, we present the experimental results over the given dataset to have a performance comparison with other conventional data analytical schemes (given as follow).

Three-sigma Rule (TSR): TSR is a typical anomaly detection method. Different from data preprocessing as presented in Section IV-A, we use it as a baseline method to detect whether an electricity customer is an electricity thief. The electricity-theft detection in TSR can be expressed as follows,

$$\text{TSR}(\theta) = \frac{\sum_{i=1}^{n} I(x_i, \mathbf{x}, \theta)}{n}, \tag{15}$$

where $I(x_i, \mathbf{x}, \theta)$ is an indicator function defined as follows,

$$I(x_i, \mathbf{x}, \theta) = \begin{cases} 1 & \text{if } x_i > \text{avg}(\mathbf{x}) + \theta \cdot \text{std}(\mathbf{x}), \\ 0 & \text{otherwise}, \end{cases} \tag{16}$$

where x_i is the i-th value of $\mathbf{x}$, n is the length of $\mathbf{x}$, θ is the outlier threshold (we typically choose $\theta = 1, 2, 3, 4$).

Logistic Regression (LR): This method is a basic model in binary classification, which is equivalent to one layer of neural network with sigmod activation function.

Random Forest (RF): In the previous study [42], decision tree was used to identify power quality disturbances. Random forest model is essentially an integration of multiple decision trees. Compared with a single decision tree, random forest model can achieve better performance while maintaining the effective control of over-fitting.

Support Vector Machine (SVM): Many previous studies exploit SVM to infer the presence of electricity theft [17][18][19].

TABLE III
PERFORMANCE COMPARISON WITH OTHER CONVENTIONAL SCHEMES (PARAMETERS $\alpha = 90, \beta = 60, \gamma = 90$)

	Training ratio = 50%			Training ratio = 60%			Training ratio = 70%			Training ratio = 80%		
Methods	AUC	MAP@100	MAP@200	AUC	MAP@100	MAP@200	AUC	MAP@100	MAP@200	AUC	MAP@100	MAP@200
TSR (1)	0.5705	0.5056	0.5140	0.5698	0.5111	0.5140	0.5593	0.5365	0.5332	0.5676	0.5284	0.5363
TSR (2)	0.5903	0.5755	0.5577	0.5847	0.5955	0.5737	0.5720	0.5255	0.5277	0.5801	0.5764	0.5654
TSR (3)	0.5514	0.5362	0.5336	0.5526	0.4774	0.4989	0.5513	0.5281	0.5326	0.5498	0.5458	0.5266
TSR (4)	0.5069	0.4973	0.5039	0.5115	0.4988	0.4979	0.5135	0.5560	0.5383	0.5034	0.5676	0.5452
LR	0.6773	0.6442	0.5669	0.6944	0.6612	0.5746	0.6916	0.6666	0.5783	0.7060	0.6560	0.5781
SVM	0.7183	0.6862	0.5919	0.7317	0.7192	0.6071	0.7276	0.7244	0.6068	0.7413	0.7353	0.6195
RF	0.7317	0.9078	0.8670	0.7325	0.8869	0.8525	0.7372	0.9259	0.8864	0.7385	0.9054	0.8536
Wide	0.6751	0.8013	0.7675	0.6950	0.8096	0.7841	0.6866	0.8116	0.7768	0.6965	0.8096	0.7646
CNN	0.7636	0.9059	0.8835	0.7837	0.9394	0.9077	0.7779	0.9547	0.9154	0.7797	0.9229	0.8853
Wide & Deep CNN	**0.7760**	**0.9404**	**0.8961**	**0.7922**	**0.9555**	**0.9297**	**0.7860**	**0.9686**	**0.9327**	**0.7815**	**0.9503**	**0.9093**

Wide: This scheme can be regarded as a variant of our proposed Wide & Deep CNN model with the removal of the CNN component. Note that the training data for the Wide scheme is the 1-D electricity consumption data (the same as our Wide & Deep CNN model).

Convolutional Neural Network (CNN): This scheme can be regarded as a variant of our proposed Wide & Deep CNN model with the removal of the Wide component and the preservation of CNN. Note that the training data for CNN is the 2-D electricity consumption data.

Table II summarizes the parameters used for the baseline methods and the extracted features to train these models. It is worth mentioning that we only used raw data and did not artificially modify the models based on any expert domain knowledge (it is relatively fair to evaluate the learning capability of each model).

Table III presents the performance comparison of our proposed Wide & Deep CNN scheme and other schemes. It is worth noting that we randomly choose a subset of electricity consumption records according to the *training ratio*, which is defined by the ratio of the size of training samples to the size of all the samples. We conduct four groups of experiments with training ratio being 50%, 60%, 70%, and 80%, respectively. Moreover, we choose parameters $\alpha = 90, \beta = 60, \gamma = 15$ for both our Wide & Deep CNN model and CNN model (without parameter α). In each group of experiments, we evaluate the performance metrics (AUC and MAPN) for the five schemes (note that we choose $N = 100$ and $N = 200$ for MAPN).

It is shown in Table III that our proposed Wide & Deep CNN scheme performs better than conventional schemes like LR, RF, Wide, SVM and CNN in terms of AUC, MAP100 and MAP200 in all four groups of experiments. For example, Wide & Deep CNN can achieve the maximum MAP100 value with 0.9404 compared with other schemes when the training ratio is 50%. This implies that our proposed method has the higher accuracy in electricity theft detection than other conventional schemes. This improvement may owe to the integration of *memorization* and *generalization* features brought by the Wide component and the Deep CNN component, respectively.

C. Parameter Study

We then investigate the impacts of various parameters on the performance of our proposed Wide & Deep CNN scheme.

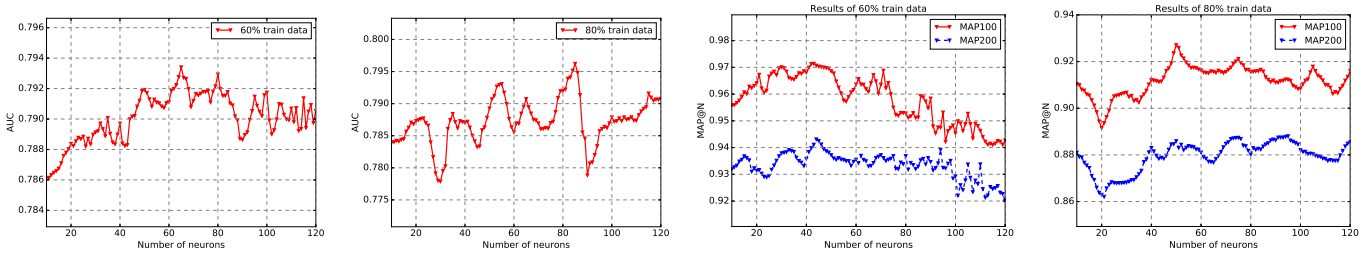
1) Effect of α: α is a parameter controlling the number of neurons in the fully-connected layer of the Wide component. To investigate the impact of α on the prediction results, we vary the values of α from 10 to 120 with the step value of 1. At the same time, we fix β = 64 and γ = 10. We conduct two groups of experiments with the training ratio with 60% or 80%, respectively.

Fig. 6 shows the experiment results. We can see from Fig. 6 that both AUC and MAP increase at first when the number of neurons (i.e., α) increases while they decrease when α is greater than a certain value. For example, when α is smaller than 50, AUC always increases in both these two groups of experiments as shown in Fig. 6 (a) and (b) and it drops when α is greater than 50 while it increases again when α is greater than 60. But, the best performance was obtained when $\alpha = 50$. This is because the Wide component may lack of enough neurons to learn from the 1-D electricity consumption data when α is too small. However, when α is too large, too many neurons will make the neural networks complicated, consequently resulting in overfitting. Therefore, there may exist a threshold on α to optimize the AUC performance. In the two groups experiments, the threshold on α is 50.

We have similar findings in the MAP performance in both two groups of experiments. Take Fig. 6 (c) and (d) as an example. MAP@N reaches the peak when $\alpha = 60$ in the first group of experiments (with training ratio 60%) and reaches the peak when $\alpha = 80$ in the second group of experiments (with training ratio 80%). This result implies that there are different thresholds on α when we optimize AUC and MAP. Therefore, it is worthwhile for us to investigate how to choose an appropriate number of neurons in the Wide component to optimize both AUC and MAP. This will be left as one of the future works.

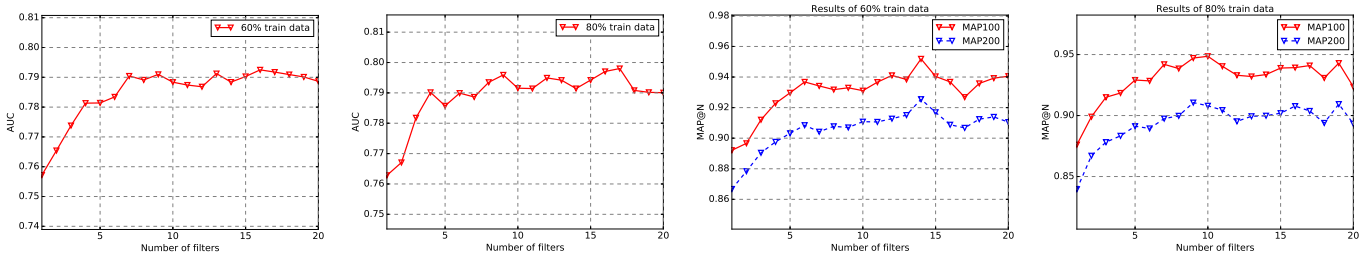
2) Effect of γ: in the CNN component of our Wide & Deep CNN method, γ is a parameter controlling the number of filters in the convolutional layer. To investigate the impact of γ on the prediction results, we vary γ from 1 to 20 with the step value of 1. Similarly, we also conduct two groups of experiments with the training ratio with 60% or 80%, respectively. Note that we also fix α = 60 and β = 60 in both the two groups of experiments.

Fig. 7 shows the experiment results. When γ increases, AUC also increases at first. However, when γ surpasses a certain threshold, AUC decreases. A similar trend can be found in MAP. This is because the CNN component may not have enough neurons to learn from the 2-D electricity consumption data when γ is small while too large value of γ may cause overfitting. Similar to γ, there may exist different threshold



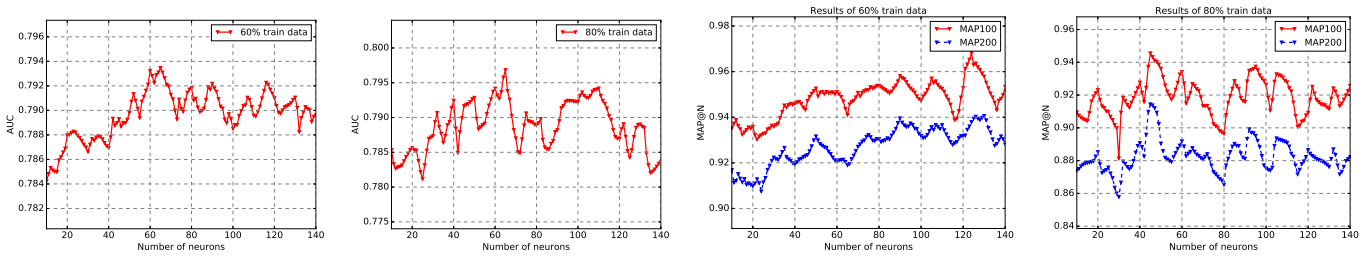
(a) AUC results with training ratio = 60% (b) AUC results with training ratio = 80% (c) MAP results with training ratio = 60% (d) MAP results with training ratio = 80%

Fig. 6. Effect of α. System parameters: γ = 10, β = 64, no. of neurons from 10 to 120 with the step value of 1.



(a) AUC results with training ratio = 60% (b) AUC results with training ratio = 80% (c) MAP results with training ratio = 60% (d) MAP results with training ratio = 80%

Fig. 7. Effect of γ. System parameters: α = 60, β = 60, no. of filters from 1 to 20 with the step value of 1.



(a) AUC results with training ratio = 60% (b) AUC results with training ratio = 80% (c) MAP results with training ratio = 60% (d) MAP results with training ratio = 80%

Fig. 8. Effect of β. System parameters: α = 60, γ = 15, no. of neurons from 10 to 140 with the step value of 1.

values of γ to optimize AUC and MAP.

3) Effect of β: β is a parameter controlling the number of neurons in the fully-connected layer in the CNN component immediately following the pooling layer. To investigate the impact of β on the prediction results, we vary the values of β from 10 to 140 with the step value of 1. At the same time, we fix α = 60 and γ = 15. We also conduct two groups of experiments with the similar settings on the training ratio with 60% or 80%, respectively.

Fig. 8 shows experiment results. We have the similar findings to those in investigating the effect of α. In fact, β plays a similar role in the CNN component like α in the Wide component. So, to avoid the repetition, we do not explain the experiment results of β in detail. However, it is worth mentioning that the range of β is usually different from that of α since the input data size for the fully-connected layer in the CNN component is different from that in the Wide component.

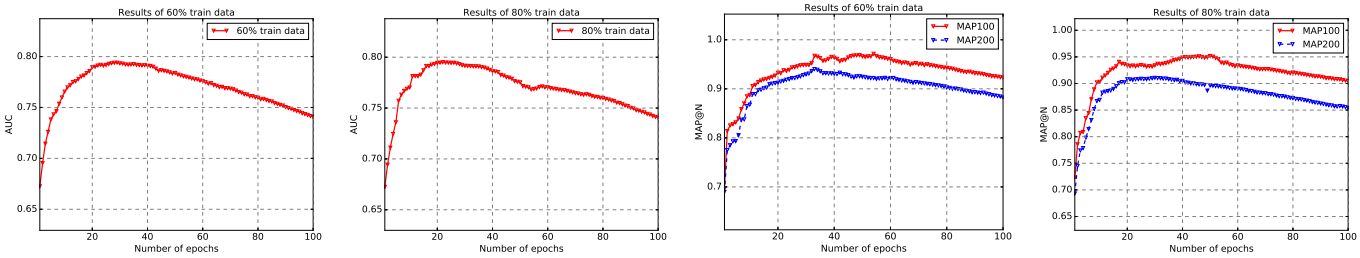
4) Effect of R: We choose R to control the number of layers in the Deep CNN component. To investigate the impact of R on the prediction results, we vary the values of R from 1 to 5 with the step value of 1. Table IV shows experiment results. Note that we choose a general setting with the training ratio 80% in CNN. It is shown in Table IV that AUC of Wide & Deep CNN increases with the increased number of layers. MAP@N has the upward trend similar to AUC with the increased number of layers. Overall, the increased number of layers in Wide & Deep CNN can improve the prediction performance. The performance improvement mainly owes to the effect that Deep CNN can capture features of 2-D data better with the increased number of layers.

TABLE IV
EFFECT OF R

	Training ratio = 80%		
# of layers R	AUC	MAP@100	MAP@200
1	0.7815	0.9190	0.8674
2	0.7872	0.9353	0.9051
3	0.7890	0.9449	0.9034
4	0.7923	0.9524	0.9080
5	0.8001	0.9565	0.9128

D. Convergence Analysis

In our Wide & Deep CNN method, the epoch is a parameter controlling the train round. An epoch is defined by one forward pass and one backward pass of all training samples. We choose the similar settings like parameter study to investigate the impact of the epoch. In particular, we vary the epoch values from 10 to 100 with the step value of 1 and we fix α = 60,



(a) AUC results with training ratio = 60% (b) AUC results with training ratio = 80% (c) MAP results with training ratio = 60% (d) AUC results with training ratio = 80%

Fig. 9. Convergence analysis. System parameters: α = 60, γ = 15, β = 120, no. of epochs from 10 to 100 with the step value of 1.

γ = 15, β = 120. Similarly, we also conduct two groups of experiments with different training ratio (60% or 80%).

Fig. 9 presents the results (in terms of AUC and MAP). We can see that like the parameter study, when the epoch value increases, both AUC and MAP increase at first. But after a certain threshold on the epoch, both AUC and MAP drop while they increase again later. This phenomenon can be explained as follows. When we choose a smaller epoch value, it may be not enough to let our Wide & Deep CNN system learn from both 1-D and 2-D data. However, it may cause overfitting when we choose a larger epoch value. Therefore, there also exists a threshold on the epoch value to optimize the training procedure in our Wide & Deep CNN. For example, the best performance was achieved when the number of epochs reaches 30 when the training ratio is 60%.

VI. Conclusion and Future Work

In this paper, we propose a Wide & Deep CNN model to detect electricity theft in smart grids. In particular, our Wide & Deep CNN model consists of the Wide component and the Deep CNN component; it gains the benefits of memorization and generalization brought by the Wide component and the Deep CNN component, respectively. We conduct extensive experiments on realistic electricity consumption data released by State Grid Corporation of China (SGCC), the largest electricity supply company in China. The experiment results show that our proposed Wide & Deep CNN outperforms existing methods, such as linear regression, support vector machine, random forest and CNN. In fact, the proposed Wide & Deep CNN model is quite general; it can be applied to other scenarios, especially for industrial applications. For example, indoor marihuana growing companies often steal electricity from the power grid [43]. Since it consumes extremely high amounts of electricity to grow marihuana, the abnormal electricity usage patterns can be captured by the proposed Wide & Deep CNN model.

References

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

[2] S. McLaughlin, B. Holbert, A. Fawaz, R. Berthier, and S. Zonouz, "A multi-sensor energy theft detection framework for advanced metering infrastructures," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1319–1330, 2013.

[3] "Smart meters help reduce electricity theft, increase safety," https://www.bchydro.com/news/conservation/2011/smart_meters_energy_theft.html, BC Hydro Inc., Tech. Rep., March 2011.

[4] H. Jiang, K. Wang, Y. Wang, M. Gao, and Y. Zhang, "Energy big data: A survey," *IEEE Access*, vol. 4, pp. 3844–3861, 2016.

[5] X. Yu and Y. Xue, "Smart grids: A cyber–physical systems perspective," *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1058–1070, 2016.

[6] Y. Liu, C. Yuen, R. Yu, Y. Zhang, and S. Xie, "Queuing-based energy consumption management for heterogeneous residential demands in smart grid," *IEEE Transactions on Smart Grid*, vol. 7, no. 3, pp. 1650–1659, 2016.

[7] K. Wang, C. Xu, Y. Zhang, S. Guo, and A. Zomaya, "Robust big data analytics for electricity price forecasting in the smart grid," *IEEE Transactions on Big Data*, 2017.

[8] Y. Wu, X. Tan, L. Qian, D. H. Tsang, W.-Z. Song, and L. Yu, "Optimal pricing and energy scheduling for hybrid energy trading market in future smart grid," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 6, pp. 1585–1596, 2015.

[9] M. H. Yaghmaee, M. Moghaddassian, and A. Leon-Garcia, "Autonomous two-tier cloud-based demand side management approach with microgrid," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1109–1120, 2017.

[10] B. C. Costa, B. L. A. Alberto, A. M. Portela, W. Maduro, and E. O. Eler, "Fraud detection in electric power distribution networks using an ann-based knowledge-discovery process," *International Journal of Artificial Intelligence & Applications*, vol. 4, no. 6, pp. 17–21, 2013.

[11] J. I. Guerrero, C. Leon, I. Monedero, F. Biscarri, and J. Biscarri, "Improving knowledge-based systems with statistical techniques, text mining, and neural networks for non-technical loss detection." *Knowledge-Based Systems*, vol. 71, pp. 376–388, 2014.

[12] C. C. Ramos, A. N. Souza, G. Chiachia, A. X. Falcao, and J. P. Papa, "A novel algorithm for feature selection using harmony search and its application for non-technical losses detection," *Computers & Electrical Engineering*, vol. 37, no. 6, pp. 886–894, 2011.

[13] L. A. P. Junior, C. C. O. Ramos, D. Rodrigues, D. R. Pereira, A. N. de Souza, K. A. P. da Costa, and J. P. Papa, "Unsupervised non-technical losses identification through optimum-path forest," *Electric Power Systems Research*, vol. 140, pp. 413–423, 2016.

[14] B. Khoo and Y. Cheng, "Using rfid for anti-theft in a chinese electrical supply company: A cost-benefit analysis," in *2011 IEEE Conference on Wireless Telecommunications Symposium (WTS)*. IEEE, 2011, pp. 1–6.

[15] S. Ngamchuen and C. Pirak, "Smart anti-tampering algorithm design for single phase smart meter applied to ami systems," in *2013 IEEE Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. IEEE, 2013, pp. 1–6.

[16] K. Dineshkumar, P. Ramanathan, and S. Ramasamy, "Development of arm processor based electricity theft control system using gsm network," in *2015 IEEE International Conference on Circuit, Power and Computing Technologies (ICCPCT)*. IEEE, 2015, pp. 1–6.

[17] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed, and M. Mohamad, "Nontechnical loss detection for metered customers in power utility using support vector machines," *IEEE transactions on Power Delivery*, vol. 25, no. 2, pp. 1162–1171, 2010.

[18] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, "Support vector machine based data classification for detection of electricity theft," in *Power Systems Conference and Exposition (PSCE)*. IEEE, 2011, pp. 1–8.

[19] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed, and F. Nagi, "Improving svm-based nontechnical loss detection in power utility using the fuzzy inference system," *IEEE Transactions on power delivery*, vol. 26, no. 2, pp. 1284–1285, 2011.

[20] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1–15:58, Jul. 2009.

[21] S.-Y. Ji, B.-K. Jeong, S. Choi, and D. H. Jeong, "A multi-level intrusion detection method for abnormal network behaviors," *Journal of Network and Computer Applications*, vol. 62, pp. 9 – 17, 2016.

[22] G. van Capelleveen, M. Poel, R. M. Mueller, D. Thornton, and J. van Hillegersberg, "Outlier detection in healthcare fraud: A case study in the medicaid dental domain," *International Journal of Accounting Information Systems*, vol. 21, pp. 18 – 31, 2016.

[23] C. Feng, T. Li, and D. Chana, "Multi-level anomaly detection in industrial control systems via package signatures and lstm networks," in *Dependable Systems and Networks (DSN), 2017 47th Annual IEEE/IFIP International Conference on*. IEEE, 2017, pp. 261–272.

[24] C. Alcaraz, L. Cazorla, and G. Fernandez, "Context-awareness using anomaly-based detectors for smart grid domains," in *CRiSIS 2014*. Springer, 2014, pp. 17–34.

[25] B. Rossi, S. Chren, B. Buhnova, and T. Pitner, "Anomaly detection in smart grid data: An experience report," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct 2016, pp. 2313–2318.

[26] T. Andrysiak, Ł. Saganowski, and P. Kiedrowski, "Anomaly detection in smart metering infrastructure with the use of time series analysis," *Journal of Sensors*, vol. 2017, 2017.

[27] V. J. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85–126, Oct 2004.

[28] J. E. Cabral, O. P. Joao, and A. M. Pinto, "Fraud detection system for high and low voltage electricity consumers based on data mining," in *Power & Energy Society General Meeting*. IEEE, 2009, pp. 1–5.

[29] C. Leon, F. Biscarri, I. Monedero, J. I. Guerrero, J. Biscarri, and R. Millan, "Integrated expert system applied to the analysis of non-technical losses in power utilities," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10 274–10 285, 2011.

[30] R. R. Wilcox, *Introduction to robust estimation and hypothesis testing*. Academic press, 2011.

[31] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Electricity Theft Detection in AMI Using Customers' Consumption Patterns," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 216–226, Jan 2016.

[32] A. Shiri, M. Afshar, A. Rahimi-Kian, and B. Maham, "Electricity price forecasting using support vector machines by considering oil and natural gas price impacts," in *2015 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, 2015.

[33] K. Wang, C. Xu, and S. Guo, "Big data analytics for price forecasting in smart grids," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016.

[34] C. Genes, I. Esnaola, S. M. Perlaza, L. F. Ochoa, and D. Coca, "Recovering missing data via matrix completion in electricity distribution systems," in *Signal Processing Advances in Wireless Communications (SPAWC), 2016 IEEE 17th International Workshop on*, 2016.

[35] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[36] H.-T. Cheng *et al.*, "Wide & deep learning for recommender systems," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. ACM, 2016, pp. 7–10.

[37] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *Neural Networks (IJCNN), 2017 International Joint Conference on*. IEEE, 2017, pp. 1578–1585.

[38] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *International Conference on Learning Representations*, 2016.

[39] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch." *Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[40] J. Davis and G. Mark, "The relationship between precision-recall and roc curves," in *Proceedings of The 23rd International Conference on Machine Learning*. ACM, 2006, pp. 233–240.

[41] A. Turpin and F. Scholer, "User performance versus precision measures for simple search tasks," in *Proceedings of the 29th annual international ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2006, pp. 11–18.

[42] M. Biswal and P. K. Dash, "Measurement and classification of simultaneous power signal patterns with an s-transform variant and fuzzy decision tree," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 1819–1827, 2013.

[43] J. Diplock and D. Plecas, "The increasing problem of electrical consumption in indoor marihuana grow operations in british columbia," https://ufv.ca/media/assets/criminology/Electrical-Consumption-in-Indoor-MGOs-in-BC-2011.pdf, University of the Fraser Valley, Tech. Rep., 2011.

Zibin Zheng is an associate professor at Sun Yat-sen University, Guangzhou, China. He received Ph.D. degree from The Chinese University of Hong Kong in 2011. He received ACM SIGSOFT Distinguished Paper Award at ICSE'10, Best Student Paper Award at ICWS'10, and IBM Ph.D. Fellowship Award. His research interests include services computing, software engineering, and blockchain.

Yatao Yang received his B.S degree in Zhengzhou University, Zhengzhou, China, in 2014. He got his M.S degree in Sun Yat-Sen University, Guangzhou, China, in 2017. He is currently a Ph.D. degree candidate in School of Data and Computer Science, Sun Yat- Sen University. His research interests include machine learning, data mining, and service computing.

Xiangdong Niu received his B.S degree in Northwest University, Xi'an, China, in 2013. He is currently a M.S degree candidate in School of Data and Computer Science, Sun Yat-sen University. His research interests include service computing and data mining.

Hong-Ning Dai is an Associate Professor in Faculty of Information Technology at Macau University of Science and Technology. He obtained the Ph.D. degree in Computer Science and Engineering from Department of Computer Science and Engineering at the Chinese University of Hong Kong in 2008. He also holds visiting positions at Department of Computer Science and Engineering, The Hong Kong University of Science and Technology and School of Electrical Engineering and Telecommunications, the University of New South Wales, respectively. His research interests include wireless networks, mobile computing, and distributed systems.

Yuren Zhou is a professor at School of Data and Computer Science, Sun-Yat Sen University. He received the B.Sc. degree in the mathematics from Peking University, Beijing, China, in 1988, the M.Sc. degree in mathematics and the Ph.D. degree in computer science from Wuhan University, Wuhan, China, in 1991 and 2003, respectively. His current research interests are focused on evolutionary computation and data mining.