energies

MDPI

Article

A Novel Electricity Theft Detection Scheme Based on Text Convolutional Neural Networks

Xiaofeng Feng [1], Hengyu Hui [2,*], Ziyang Liang [2], Wenchong Guo [1], Huakun Que [1], Haoyang Feng [1], Yu Yao [2], Chengjin Ye [2] and Yi Ding [2]

[1] Metrology Center of Guangdong Power Grid Corporation, Guangzhou 510080, China; ucihqtep@163.com (X.F.); wenchong1025@163.com (W.G.); quehuakun@126.com (H.Q.); fenghy111@163.com (H.F.)

[2] College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China; liangziyang@zju.edu.cn (Z.L.); zjuyaoyu@zju.edu.cn (Y.Y.); yechenjing@zju.edu.cn (C.Y.); yiding@zju.edu.cn (Y.D.)

* Correspondence: huihengyu@zju.edu.cn; Tel.: +86-136-2839-5130

Received: 30 September 2020; Accepted: 30 October 2020; Published: 3 November 2020

check for updates

Abstract: Electricity theft decreases electricity revenues and brings risks to power usage's safety, which has been increasingly challenging nowadays. As the mainstream in the relevant studies, the state-of-the-art data-driven approaches mainly detect electricity theft events from the perspective of the correlations between different daily or weekly loads, which is relatively inadequate to extract features from hours or more of fine-grained temporal data. In view of the above deficiencies, we propose a novel electricity theft detection scheme based on text convolutional neural networks (TextCNN). Specifically, we convert electricity consumption measurements over a horizon of interest into a two-dimensional time-series containing the intraday electricity features. Based on the data structure, the proposed method can accurately capture various periodical features of electricity consumption. Moreover, a data augmentation method is proposed to cope with the imbalance of electricity theft data. Extensive experimental results based on realistic Chinese and Irish datasets indicate that the proposed model achieves a better performance compared with other existing methods.

Keywords: data-driven approaches; electricity theft detection; smart meters; text convolutional neural networks (TextCNN); time-series classification

1. Introduction

Electricity theft can be defined as the behavior of illegally altering an electric energy meter to avoid billing. This illegal behavior not only severely disrupts the normal utilization of electricity, but also causes huge economic losses to power systems. At the same time, the unauthorized modification of lines or meters easily leads to accidents such as power failures and fires, and poses a serious threat to the safety of the relevant power system [1,2]. According to the research released by an intelligence firm northeast group, llc in January 2017, electricity theft and other non-technical losses have rendered over $96 billion in losses per year globally [3]. State Grid Hunan Electric Power Company, China reported that nearly 40% of electrical fires and 28% of electric shock casualties are caused by electricity theft [4]. Therefore, it is necessary to develop effective techniques for electricity theft detection and ensure the security and economic operation of power system.

The electricity theft detection technologies can be divided into three categories: the network-oriented method, the data-oriented method and a hybrid-oriented method that mixes the previous two methods [5]. Network-oriented and hybrid-oriented approaches usually require the network topology [6,7] and even the installation of additional devices [8]. It is difficult to implement these approaches widely, because the network topology may be unattainable due to security concerns and the installation of addition devices is costly. Data-oriented approaches only focus on the data provided by smart meters and have no requirements of the network topology or additional devices, which helps with improving the cost-effectiveness for suspected electricity theft judgment and detection. Therefore, data-oriented approaches have been widely applied to electricity theft detection in recent years [9,10].

At present, there are two typical data-oriented methods to detect electricity theft: support vector machines (SVM) and neural networks. In [11], they proposed a SVM-based approach that uses customer consumption data to expose abnormal behavior and identify suspected thieves. The authors in [12,13] combined SVM and a fuzzy inference system to detect electricity theft. In [14], a comprehensive top-down scheme based on decision trees and SVM was proposed. The two-level data processing and analysis approach can detect and locate electricity theft at every level in power transmission and distribution. The authors in [15] proposed an ensemble approach combining the adaptive boosting algorithm and SVM.

More and more researchers are utilizing neural networks to detect electricity theft due to their effectiveness. In [16], a long short-term memory (LSTM) and bat-based random under-sampling boosting (RUSBoost) approach is proposed. The LSTM and bat-based RUSBoost are applied to detect abnormal patterns and optimize parameters, respectively. In [17], a method based on the wide and deep convolutional neural network (CNN) model is proposed. The deep CNN component can identify the periodicity of electricity consumption and the wide component can capture the global characteristics of electricity consumption data. The authors in [18] combined CNN and LSTM to detect electricity theft from the power consumption signature in time-series data. In [10], an end-to-end hybrid neural network is proposed, which can analyze daily energy consumption data and non-sequential data, such as geographic information.

The above methods have paved the way for building the structures of networks and dedicate to improving electricity theft detection's accuracy. However, they mainly focus on the daily or weekly electricity consumption patterns. As a result, if these methods are applied to hourly or more frequent electricity consumption data, their accuracy will decrease. This is because they fail to capture the intraday electricity consumption pattern, for example, the correlation between the electricity consumption at the same time on different days. In practice, some illegal users commit electricity theft for part of the day. Specifically, those who have precedent technology and large electricity demands (such as industrial electricity thieves) prefer to commit the crime during some specific hours after considering electricity prices, monitoring periods and the risk of being caught comprehensively. Meanwhile, new kinds of attacks such as interception communication and false data injection [19,20] make it easier to commit such crime. In a case of electricity theft caught by State Grid Shandong Electric Power Company, China, the illegal user had normal daily electricity consumption. However, he confused the metering time of his smart meter to avoid peak electricity tariffs [21]. In this way, his abnormal electricity consumption pattern can only be reflected in the intraday data. In paper [22], several possible attack models are proposed to confuse the metering time and commit electricity theft targeting time-based pricing. Therefore, it is necessary to construct an electricity theft detection scheme that can not only capture the daily features but the intraday features.

In order to better extract the periodical features from days and more frequent time periods, we utilize a two-dimensional grid structure for the raw input data in this paper. Based on the data structure, we propose a text convolutional neural network (TextCNN) to detect electricity theft. The main contributions of the proposed model include:

(1) We analyze the electricity data structure and transform it into a two-dimensional time-series. This structure carries the complete power consumption information of users, which means the consumption patterns of various time scales, such as the electricity consumption at the same period on different days and the daily consumption of different days.

(2) We propose a novel electricity theft detecting method based on TextCNN. The proposed method can extract features of different time scales from two-dimensional time-series. To improve the

accuracy and efficiency of training and detection, we designed our detection network based on TextCNN. To test the performance, we implemented extensive experiments on the residential and industrial datasets from a province in China and the public Irish residential dataset.

(3) We propose the data augmentation method to expand the training data in view of the shortage of electricity theft samples. Experimental analysis indicates that the method can improve the detection accuracy effectively with a proper augmentation process.

The remainder of this paper is organized as follows. The methodologies of data construction and CNN construction for electricity consumption data are described in Section 2. Section 3 proposes the neural network structure based on TextCNN and the data augmentation method. Then, the details of experimental datasets and methods used for comparison and metrics are given in Section 4. Section 5 presents the performance and superiority of the proposed model, analyzes the parameters and discuss the effectiveness of the data augmentation method. Finally, Section 6 concludes this paper.

2. Methodology

In this section, we introduce the characteristics of the electricity consumption data and then compare several data structures regarding their advantages and disadvantages. Finally, we introduce TextCNN and the reason why it is suitable for the two-dimensional time-series.

2.1. Data Structure Analysis

Smart meters can collect the electricity consumption data at a high frequency, such as once an hour. The datasets can be expressed as:

$$D_n = \left\{x_{h_1}^{d_1}, x_{h_2}^{d_1}, \ldots, x_{h_{24}}^{d_1}, x_{h_1}^{d_2}, \ldots, x_{h_j}^{d_i}\right\} \tag{1}$$

where D_n represents the data of user n. $x_{h_j}^{d_i}$ is the value recorded by smart meters during time h_j on day d_i.

Most studies focus on periodical features of daily or weekly consumption patterns to detect electricity theft. Therefore, they merge the data of one day into one value and utilize the one-dimensional data structure or its variant, as shown in Figure 1. The datasets can be further expressed as:

$$D_n = \left\{x_{d_1}, x_{d_2}, \ldots, x_{d_i}\right\} \tag{2}$$

where x_{d_i} represents the total amount of electricity consumption on day d_i.

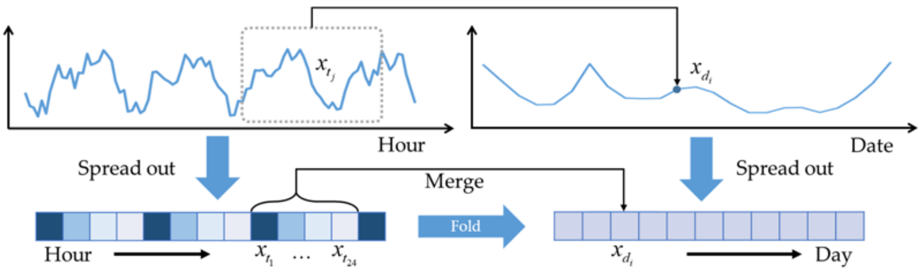


Figure 1. One-dimensional data structure of electricity consumption data.

In this way, they neglect the intraday electricity change and fail to capture the intraday features. In this paper, we construct the data into a two-dimensional grid, which is suitable for feature extraction from not only different days but different time periods. The two-dimensional grid can be expressed as:

$$D_n = \begin{bmatrix} x_{h_1}^{d_1} & x_{h_1}^{d_2} & \cdots & x_{h_1}^{d_i} \\ x_{h_2}^{d_1} & x_{h_2}^{d_2} & \cdots & x_{h_2}^{d_i} \\ \vdots & \vdots & \ddots & \vdots \\ x_{h_{24}}^{d_1} & x_{h_{24}}^{d_2} & \cdots & x_{h_{24}}^{d_i} \end{bmatrix} \tag{3}$$

The columns in Equation (3) represent the electricity consumption data of 24 h a day. In fact, smart meters may collect data more frequently, and may collect varieties of information, such as three-phase voltages and currents, power factors and so on. In this manner, in order to simplify the expression of Equation (3), we utilize the following column vector to represent the amount of data on day d_i:

$$\mathbf{x}_{d_i} = \left\{x_1, x_2, \ldots, x_j, \ldots, x_F\right\}^{\mathbf{T}} \tag{4}$$

where F is the number of data from one day. So far, the dataset of one user can be expressed as:

$$D_n = \left\{\mathbf{x}_{d_1}, \mathbf{x}_{d_2}, \ldots, \mathbf{x}_{d_i}\right\} \tag{5}$$

Further, we use Figure 2 to explain the above two-dimensional structure. In the left figure, an individual curve represents the data x_j on different days and the cluster of curves demonstrates the daily electricity consumption. Then the expansion of the cluster is a grid, as shown in the right figure, which is also formulated as Equation (5). The height of the grid represents the number of data points from one day points. The length of the grid represents the number of days. In other words, the grid of electricity consumption data is a two-dimensional time-series.

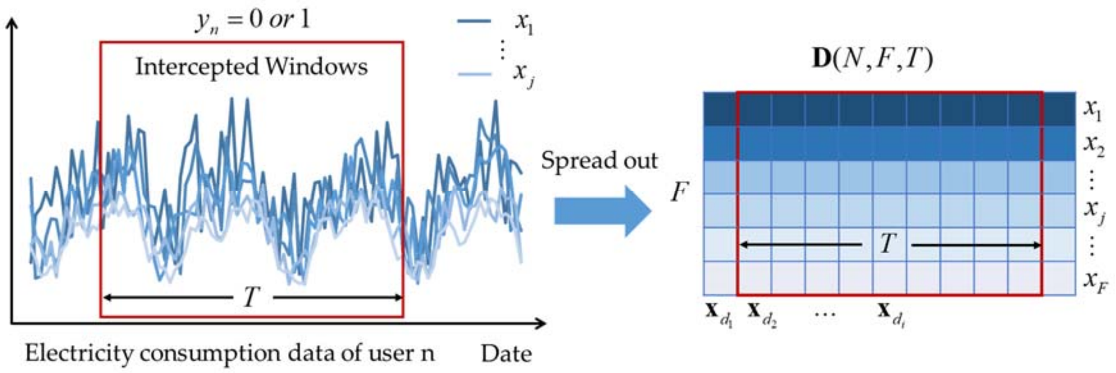


Figure 2. Two-dimensional data structure of electricity consumption data.

To extract the consumption patterns of different users, we utilize the same intercepted window to consecutively intercept different users' data. Then we can obtain a series of two-dimensional grids with the same length and height. For user n, we use y_n to label the intercepted window of the time-series to judge whether it is electricity theft or not, as shown in Figure 2. Then, we build a nonlinear map function from an input time-series to predict a class label y_n formula:

$$y_n = f(\mathbf{x}_{d_i}, T)\ d_i \in T \tag{6}$$

where T is the length of the intercepted window and $f(\cdot)$ is the key nonlinear function we aim to learn.

In order to conveniently express this data structure in CNN, we use $\mathbf{D}(N, F, T)$ to represent the intercepted segments, where N is the number of samples. For an individual data $\mathbf{D}(F, T)$ in the dataset $\mathbf{D}(N, F, T)$, the classification function $f(\cdot)$ needs learning. So far, we have constructed the two-dimensional structure that maintains the full information of the raw data and transforms the electricity theft detecting problem into the classification of time-series. Based on the two-dimensional time-series structure, we utilize TextCNN to learn the classification function.

2.2. CNN Structure Analysis

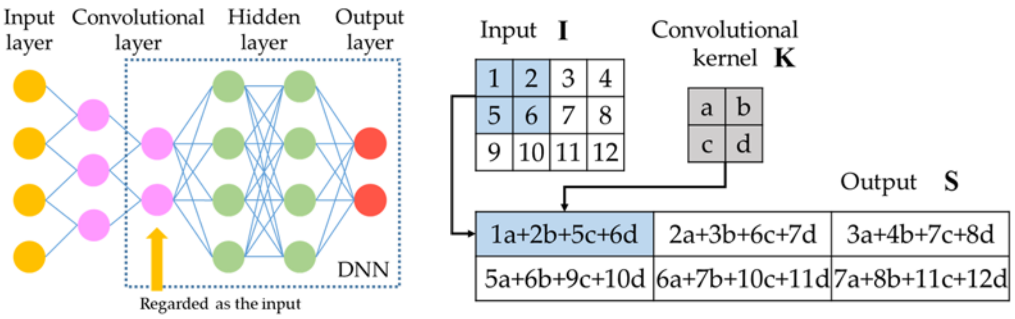
CNN specializes in processing data with a grid-like structure [23]. For different input data types, the structure of CNN should be selected further to achieve effectiveness—TextCNN, RCNN, etc. [24–26]. Considering the above two-dimensional time-series, we focus on TextCNN in this research. TextCNN is widely used in natural language processing (NLP) fields such as text classification, emotion analysis and sensitivity analysis for its simple structure and effectiveness [27,28].

2.2.1. Basic Introduction to CNN

The normal multilayered neural networks, which are also called deep neural networks (DNN), consist of input layers, hidden layers and output layers. CNN has an additional convolutional layer, as shown in Figure 3a. The discrete convolution is the key operation in convolutional layers. As shown in Figure 3b, we use a 2 × 2 kernel as an example to illustrate the discrete convolution. The input **I** has a value in each grid. Then, a two-dimensional kernel function $\mathbf{K} \in \mathbb{R}^{2\times 2}$ is used to extract features. The output **S** of the convolution is:

$$\mathbf{S}(i,j) = \sum_{k_i=0}^{1}\sum_{k_j=0}^{1}\mathbf{I}\big(i+k_i, j+k_j\big)\mathbf{K}\big(k_i,k_j\big) \tag{7}$$

Equation (7) and Figure 3b together illustrate that convolutional kernels map the neighboring information of the input into the output. Therefore, compared with DNN, CNN has an advantage of considering the information in the small neighborhoods, which is a crucial future in the classification of two-dimensional data, as the neighboring grids usually carry related information [29,30]. For example, if we regard **I** as a black and white picture, the kernels can efficiently extract features, such as edges, angles and shapes from neighboring pixels.



(**a**) (**b**)

Figure 3. Diagrams of the CNN structure and the discrete convolution: (**a**) CNN structure; (**b**) discrete convolutions.

2.2.2. Differences between CNN and TextCNN

The kernel size is the main difference between CNN and TextCNN. As shown in Figure 4a, we use height and length to describe the size of a two-dimensional kernel. The commonly used kernel size in CNN is 3 × 3 [31,32], while in TextCNN the height of kernels is always equal to that of input data [27]. This is because for text classification, the most significant thing is to efficiently capture the internal features of an individual word and the correlations between multiple words. As shown in Figure 4a, the convolutional kernels are sliding windows with the same height as a single world. The kernel will only move in the length direction, so each time the kernel will slide over a complete word.

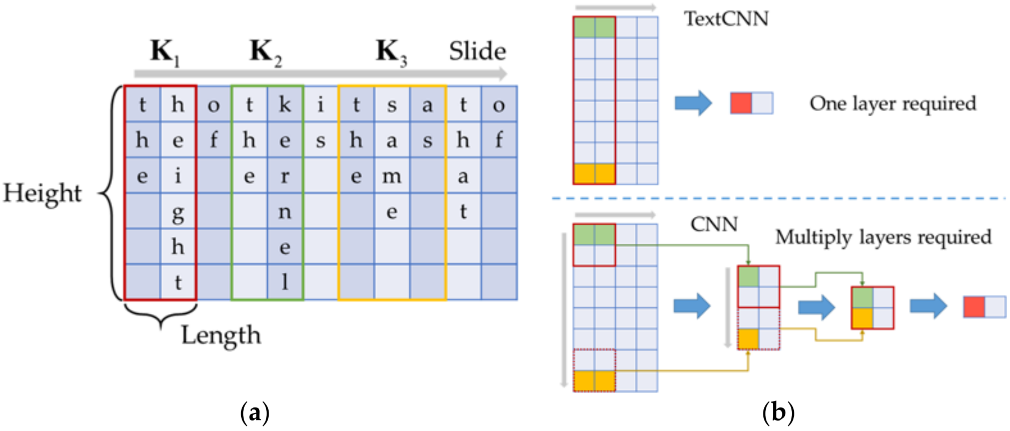


Figure 4. Characteristics of CNN: (**a**) feature schematic diagram of TextCNN's kernels; (**b**) differences between CNN and TextCNN.

The influences of different kernel sizes on the network are shown in Figure 4b. In order to capture the association between the green grids and the yellow grids, TextCNN requires only one convolutional layer, while CNN requires three convolutional layers. Therefore, TextCNN simplifies the structure of the neural network and reduces the parameters that require manual intervention. In this manner, the efficiency and effectiveness of capturing the internal features of a word and the correlations between multiple words are guaranteed.

In electricity theft detection, we aim to capture the features from the data correlations of weeks, days, hours and even more frequent time periods. Analogously, the intraday feature of electricity consumption is similar to the association between the green grids and the yellow grids in Figure 4b, and the multi-day correlations are extracted by different kernels, such as $\mathbf{K}_1$, $\mathbf{K}_2$ and $\mathbf{K}_3$ in Figure 4a. Therefore, to efficiently extract features of electricity consumption data, we propose a neural network based on TextCNN for the classification of two-dimensional time-series.

3. Proposed Approach

In this section, we propose our electricity theft detection scheme. We introduce the data preprocess at first. Then, we propose a neural network structure based on TextCNN, which consists of convolutional layers, pooling layers and fully-connected layers. Finally, we propose the data augmentation method to increase the amount of electricity theft data for the balance of the training dataset. The total framework of the proposed electricity theft detection is demonstrated in Figure 5.

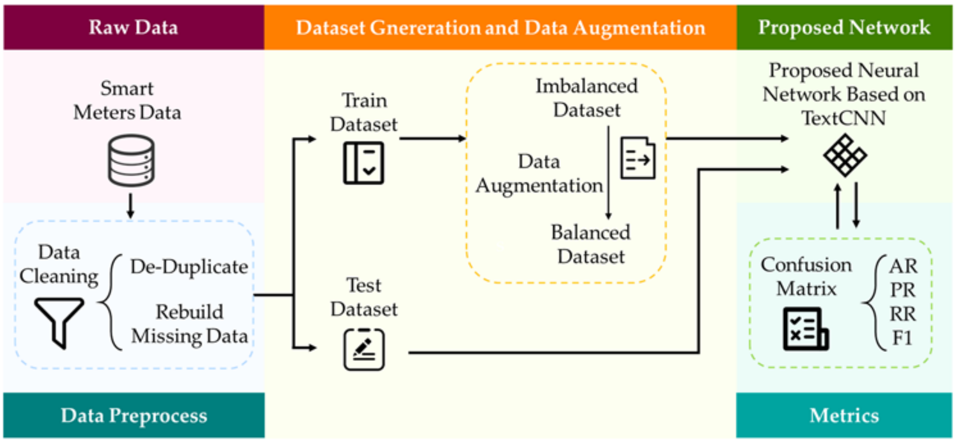


Figure 5. Proposed electricity detection scheme.

As shown in Figure 5, the raw data collected by smart meters gets through the data preprocess at first. Then, we divide it into the training dataset and the test dataset. If the training dataset is imbalanced, we utilize the proposed data augmentation method to balance it. Finally, we train the proposed network on the training dataset and validate the effect on the test dataset. The metrics used for training and testing are introduced in Section 4.3. It should be noted that the training process is supervised learning which requires labeled datasets.

3.1. Data Preprocess

During data collection, missing data, duplications and errors of electricity consumption data may occur. To avoid the adverse effects of faulty data on the electricity theft detection, reference [17] proposes an electricity data preprocessing method to recover the missing and erroneous data. Equation (8) represents the interpolation method to recover the mission data.

$$x^*_{d,t} = \begin{cases} \frac{x_{d,t-1}+x_{d,t+1}}{2} & x_{d,t} \in \mathrm{NaN}, x_{d,t-1}, x_{d,t+1} \notin \mathrm{NaN} \\ 0 & x_{d,t} \in \mathrm{NaN}, x_{d,t-1} \text{ or } x_{d,t+1} \in \mathrm{NaN} \\ x_{d,t} & x_{d,t} \notin \mathrm{NaN} \end{cases} \tag{8}$$

where $x_{d,t}$ is the electricity consumption data during time period t on day d. Additionally, NaN represents null and non-numeric character.

Moreover, the three-sigma rule of thumb [33] is used to recover the erroneous data as follows:

$$x^*_{d,t} = \begin{cases} \mathrm{avg}(\mathbf{x}_d) + 2 \cdot \mathrm{std}(\mathbf{x}_d) & \text{if } x_{d,t} > \mathrm{avg}(\mathbf{x}_d) + 2 \cdot \mathrm{std}(\mathbf{x}_d) \\ x_{d,t} & \text{otherwise} \end{cases} \tag{9}$$

where $\mathbf{x}_d$ is a vector composed of $x_{d,t}$, $\mathrm{avg}(\mathbf{x})$ and $\mathrm{std}(\mathbf{x})$ stands for the average value and standard deviation value of vector $\mathbf{x}$.

3.2. Proposed Neural Network Structure Based on TextCNN

As shown in Figure 6, the proposed neural network structure based on TextCNN is mainly composed of convolutional layers, pooling layers and fully-connected layers. The features of each layer are demonstrated in detail below.

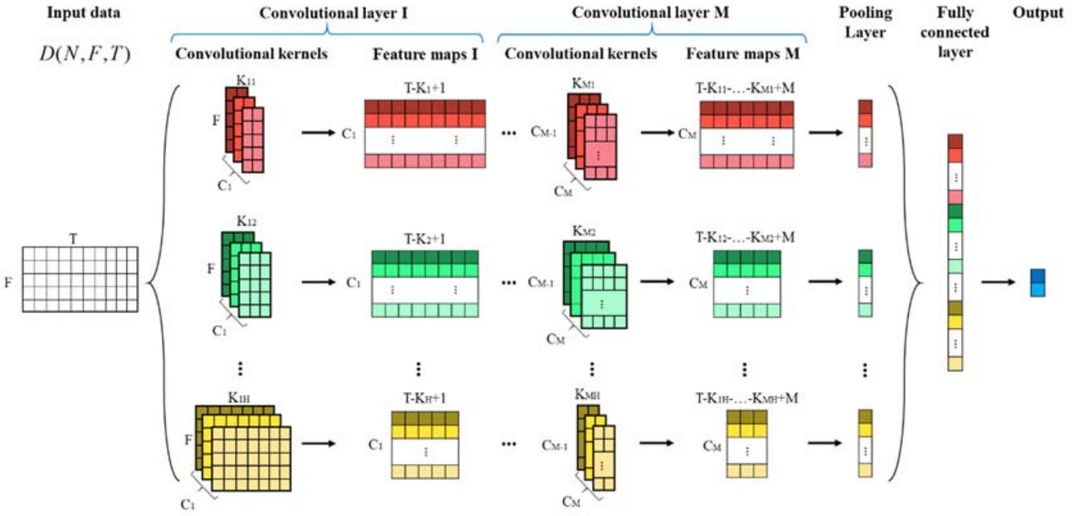


Figure 6. Proposed network structure for electricity theft detection.

3.2.1. Convolutional Layer

The proposed model has multiple convolutional layers. A convolutional layer has H different sizes of convolutional kernels. As mentioned in Section 2, in order to ensure the efficiency and effectiveness of the classification for two-dimensional time-series, the height of kernels is the same as the number of data points for one day. For convolution kernels of size H_i, $\mathbf{D}^u = (F, T)$ denotes the uth data sample. The corresponding kernel weight $\mathbf{w}_j^u \in \mathbb{R}^{F \times K}$ is used to extract features from the input data, where K is the kernel length. For example, the feature map $o_{j,i}^u$ is calculated by:

$$o_{j,i}^u = f_a\left(\mathbf{w}_j^u * \mathbf{D}^u + b_j^u\right) \tag{10}$$

where $*$ means the convolutional operation. $b_j^u \in \mathbb{R}$ is a bias term and $f_a(\cdot)$ is a nonlinear activation function such as the rectified linear unit (ReLU) function. Without the activation function, the output of the next layer is a linear function of the input of the previous layer. Additionally, it is easy to prove that no matter how many convolutional layers there are, the output is a linear combination of inputs, which means the network has no hidden layer. Therefore, activation functions can improve the effectiveness of neural networks.

There are C kernels $\left\{\mathbf{w}_1^u, \mathbf{w}_2^u, \cdots, \mathbf{w}_j^u, \cdots \mathbf{w}_C^u\right\}$ of size H_i to produce C feature maps as follows:

$$\mathbf{o}_i^u = \left[o_{1,i}^u, o_{2,i}^u, \cdots, o_{j,i}^u, \cdots, o_{C,i}^u\right]^T \tag{11}$$

After first convolution, the feature maps of kernel size H_i are represented by $\mathbf{D}_i(N, C, T - K + 1)$.

In order to extract the time features and compress the amount of data, the feature maps of the first convolutional layer should be convoluted multiple times. Thus, there are multiple convolutional layers in the proposed neural network. It is worth noting that the kernel size of the previous layer is not necessarily equal to that of the next layer. For instance, the kernel size of $\mathbf{D}_i$ in the upper layer is H_{i1}, and in the next layer is H_{i2}. H_{i1} and H_{i2} are independent of each other. After passing through these convolutional layers, the feature maps of kernel size $\{H_{i1}, H_{i2}, \cdots, H_{iM}\}$ are expressed as:

$$\mathbf{D}_i(N, C, T - K_1 - K_2 - \cdots - K_M + M) \tag{12}$$

where K_M is the kernel length of convolutional layer M.

3.2.2. Pooling Layer

After multiple convolutional operations, the data come to the pooling layer. In this paper, a max pooling layer is adopted. In the max pooling layers, only the maxima of extracted feature values are retained and all others are discarded. The max pooling layer can extract the strongest feature and discard the weaker ones. After the max pooling operation, the output is described as $\mathbf{D}_i(N, C, 1)$.

3.2.3. Fully-Connected Layer

In the fully-connected layer, the input is the stack of the pooling layer's output. Then, we use a two-class classification, the softmax activation function, to calculate the classification result which consists of two probabilities. When the probability of committing electricity theft is greater than that of being normal, the input data are labeled as electricity theft. The final output of the entire model is expressed as:

$$f_{Softmax}\left[\mathbf{D}(N, \sum_i C, 1)\right] = \mathbf{D}(N, 2, 1) \tag{13}$$

3.2.4. Parameters of the Proposed Neural Network

The main parameters of the proposed neural network are as follows. We utilize two convolutional layers to extract the features. The two convolutional layers are selected to make a balance between the accuracy and computational time. In specific, the increase of convolutional layers may improve the accuracy, but the computation burden also increases significantly with the increase of convolutional layers. Therefore, we used two convolutional layers to balance the accuracy and the efficiency in the experiments in this paper. Moreover, more layers typically means a larger number of parameters, which makes the enlarged network more prone to overfitting [29].

Each convolutional layer has multiple kernels with different sizes. Considering the characteristics of the TextCNN, the height of the kernels is same as the number of data points from one day and the lengths of kernels are 2, 3, 5 and 7. Kernels with a length of 2 or 3 can capture features from adjacent days. Additionally, kernels with lengths of 5 and 7 can capture features from the periodicity of weekday and week, respectively. Besides, in order to reduce the risk of overfitting, the dropout rate of the proposed neural network is set at 0.4.

3.3. Data Augmentation

When using CNN to cope with the classification problem, it requires a large amount of data in various categories for training to obtain a more accurate classification result. Therefore, multiple methods are used to increase the image samples in the image classification problem [34,35]. In realistic datasets, since most users do not carry out electricity theft, there are less electricity theft data compared with the normal data. The imbalance of the datasets would affect the classification result easily, which could contribute to low accuracy or overfitting. Therefore, we propose a data augmentation method to address the imbalance problem.

The data augmentation is illustrated in Figure 7. Assuming the date of electricity theft is found on day D_T, then $[D_T - T, D_T]$ is an electricity theft sample. Due to the continuation of the theft behavior, electricity theft also occurs during the time $[D_T - T - 1, D_T - 1]$. Therefore, $[D_T - T - 1, D_T - 1]$ is also an electricity theft sample. If the intercepted window slides AG times, one electricity user datum can be transformed into $AG + 1$ samples. So far, the electricity theft samples can be increased effectively through this method. It is noted that the value of AG needs to be chosen appropriately. If AG is too large, it will classify the normal data into the theft samples and affect the classification result.

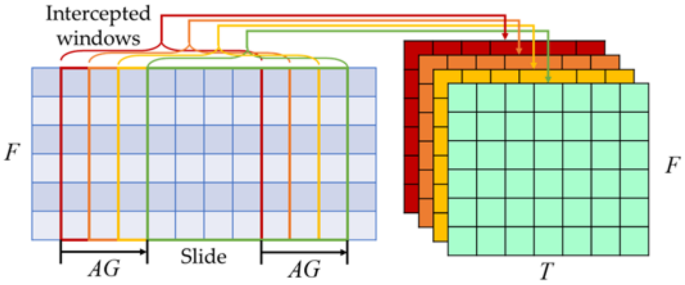


Figure 7. Diagram of the data augmentation method.

4. Experimental Settings

In this Section, we present the details of datasets. Then, we introduce several methods for comparison and the metrics to evaluate the accuracy of the classification model.

4.1. Datasets

Datasets (a) and (b) are realistic datasets from a certain province of China, containing electricity thieves and normal users. Dataset (c) is the public power data of Northern Ireland, which lacks the electricity theft data, so the electricity theft data in dataset (c) is artificially constructed.

(a) Residential user dataset

The amount of residential user data within 1277 days is quite large, so data filtering was required. There were 1063 electricity thieves; all of them were retained. The number of normal users is particularly large, reaching several million. In order to achieve a better classification effect, the ratio of electricity theft data to normal user data should be around 1:3. Therefore, the amount of normal user data finally obtained was 3564. The smart meters collected 5 data points per day, which means the number of data points from one day F was 5.

(b) Industrial user dataset

The industrial user dataset contains the electricity data of 8144 users within 1277 days, and smart meters also collected 5 data points per day, which means the number of data points from one day F was 5. Compared with the residential user dataset, the number of electricity thieves in the industrial user dataset is even smaller—only 92. The electricity thieves only occupy nearly 1% of all. Therefore, the proposed data augmentation method was used to increase the amount of the electricity theft data.

(c) Ireland residential user dataset

The dataset contains the electricity data of 5000 users within 535 days in Ireland, and smart meters collected 48 data per day (sampling every half hour), which means the number of data points from one day F was 48. These users were all normal users, so the dataset lacks electricity theft samples. As a result, we adopted a method introduced in [11] to produce electricity theft samples artificially. Since the electricity theft samples in this dataset are completely artificially generated, their number can be easily changed without using the data augmentation.

The details of each dataset are shown in Table 1.

Table 1. Datasets' information.

Datasets	(a)	(b)	(c)
Time	1 October 2015–31 March 2019	1 October 2015–31 March 2019	1 January 2009–31 December 2010
Total uses	4627	8144	5000
Normal uses	3564	8052	5000
Electricity thieves	1063	92	0

In dataset (a), the ratio of normal users to electricity thieves is 3.3:1, which satisfies the balance between positive data and negative data for training. However, in dataset (b) and dataset (c), electricity theft data needs to be augmented in order to satisfy the balance. In dataset (b), the proposed data augmentation method increases the electricity theft data in the training set to 520, and the ratio of normal data to electricity theft data in the training set is 2.2:1. In dataset (c), 1800 electricity theft samples are artificially generated, and the ratio is 2.8:1.

4.2. Baselines

Other than our proposed model, four classical models in machine learning are given for comparison. The basic parameters setting for baseline methods are summarized in Table 2.

- Logistic regression (LR). Logistic regression is a statistical model that models the probabilities for classification problems with the dependent variable being binary. It uses maximum likelihood estimation to estimate regression model coefficients that explain the relationship between input and output.
- Support vector machine (SVM). A support vector machine is a supervised learning model and can be used for classification. It uses a kernel trick to map the input into high-dimensional feature spaces implicitly. Then, SVMs construct hyperplane in high-dimensional space, and the hyperplane can be used for classification.

- Deep neural network (DNN). A deep neural network is a feedforward neural network with multilayered hidden layers. DNN can model complex non-linear relationships through the neurons in the hidden layer, which can be used for classification problem. Moreover, backpropagation algorithm is used to update the weight in DNN, because it can compute the gradient of the loss function with respect to the weights of the network efficiently.
- One-dimensional CNN (1D-CNN). The 1D-CNN is a classifier model which is similar to the proposed model. However, the user data are 1D electricity consumption data, and the dimensions of input data would be $D(N, 1, T)$. The structure of 1D CNN is the same as the proposed model mentioned in Section 3.2.

The general process of the experiments for all methods is as follows:

At first, we divide one dataset into two parts, one for training and the other for effect evaluation. The ratio of these two parts is called the training ratio. It is worth noting that positive samples and negative samples are divided separately, so the ratio of positive and negative samples in the training dataset is the same as that in the test dataset. At each training ratio, we implement ten experiments. The division of the training dataset and the test dataset is random and independent in each experiment. At last, we use the average result of these ten experiments to represent the final results.

Table 2. Parameter settings.

Baselines	Data Dimension	Parameters
LR	1-D	Penalty: L1 Solver: Liblinear Inverse of regularization strength: 1
SVM	1-D	Regularization parameter: 1.0 Kernel: RBF
DNN	1-D	Hidden layer: 3 Neurons in the hidden layer: 100, 60, 60
1D-CNN	1-D	Same as parameters of the proposed method
Proposed Method	2-D	Introduced in Section 3.2.4

4.3. Metrics

There are many ways to evaluate the classification accuracy. The evaluation metrics used in this paper are accuracy rate, precision rate, recall rate and F1.

The above four metrics were calculated based on the confusion matrix shown in Table 3.

Table 3. Confusion matrix.

Confusion Matrix		Actual	
		Negative (Normal)	**Positive (Theft)**
Classified	**Negative (normal)**	True Negative (TN)	False Negative (FN)
	Positive (theft)	False Positive (FP)	True Positive (TP)

In this paper, our purpose is to detect electricity theft. Therefore, we define electricity theft samples as positive samples, and normal samples as negative samples. Furthermore, metrics true positive (TP), true negative (TN), false positive (FP) and false negative (FN) can be obtained from the confusion matrix. TP and TN indicate that the actual attribute of the sample is the same as the classified one, which means the classification result is accurate. FP indicates that the sample is actually negative, but the classified result is positive. FN indicates that the sample is actually positive, while the classified result is negative. The contrast between actual and classified results reflects the inaccuracy of the classification model.

Accuracy rate (AR) is the proportion of correctly classified samples in all samples. It is the most intuitive and commonly used criterion to measure the classification effect of the model. The formula is as follows:

$$AR = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{14}$$

However, most samples in the training set are normal, and only a few of them committed electricity theft, which means that there are far more actual negative samples than actual positive samples. If the model classifies all actual positive samples into negative, the accuracy rate of the model will still be very high. Therefore, only using the AR criterion to evaluate the accuracy is not comprehensive.

Precision rate (PR) refers to the proportion of actual positive results in the classified positive samples, which indicates the classification accuracy in the classified positive samples. The formula is as follows:

$$PR = \frac{TP}{TP + FP} \times 100\% \tag{15}$$

Recall rate (RR) is defined as the proportion of classified positive results in the actual positive samples, which means the classification accuracy in the actual positive samples. The formula is as follows:

$$RR = \frac{TP}{TP + FN} \times 100\% \tag{16}$$

F_score is the harmonic mean of the precision rate and the recall rate, so it is more comprehensive to evaluate the accuracy. The formula is as follows:

$$F\_score = \frac{\left(\alpha^2 + 1\right) \times PR \times RR}{\alpha^2 \times (PR + RR)} \times 100\% \tag{17}$$

where α is a parameter greater than 0. In particular, when α equals one, the F_score is expressed as F1, which is the most representative criterion in common use. The formula is as follows:

$$F1 = \frac{2 \times PR \times RR}{PR + RR} \times 100\% \tag{18}$$

All in all, we construct a confusion matrix and four indicators AR, PR, RR and F1 to comprehensively consider the accuracy of the classification model. In the next section, we will analyze different models in different datasets based on the proposed metrics.

5. Results and Analysis

In this Section, we present the experimental results and analysis. We compare the performances of the proposed model with those of other methods first. Then, we study the influences of the parameters on the results. Last, we discuss the effectiveness of the proposed data augmentation method.

5.1. Performance Comparison

The performance comparison between the proposed model and other models in three datasets is demonstrated in Table 4.

The proposed model performs better than other models in different training ratios, as shown in Table 4. Take the result of a 70% training ratio as an example. The proposed model has the highest PR and RR for each dataset. However, other models had better ARs in some dataset. For example, the AR of the 1D-CNN model was the highest for dataset (a)—4.3% higher than that of the proposed model. However, F1 (which is the most comprehensive indicator of the classification performance) of the proposed model was the highest in each dataset and reached 0.757, 0.850 and 0.904 in dataset (a), dataset (b) and dataset (c), which is 20.1%, 15.2% and 8.9% higher than the second-place model respectively. Meanwhile, the proposed model performed better with the increase in the training ratio.

For example, in dataset (c), the F1 increased from 0.759 to 0. 896 as the training ratio increased from 50 to 80%.

Table 4. Results on different datasets with different models.

Training = 50%												
Model	**Dataset (a)**				**Dataset (b)**				**Dataset (c)**			
	AR	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**
LR	0.851	0.623	0.488	0.547	0.577	0.487	0.442	0.463	0.867	0.706	0.827	0.762
SVM	0.694	0.733	0.022	0.042	0.523	1.000	0.023	0.045	0.793	0.886	0.223	0.356
DNN	0.843	0.596	0.429	0.487	0.514	0.384	0.349	0.366	0.844	0.655	0.543	0.575
1D-CNN	0.871	0.689	0.439	0.536	0.714	0.913	0.488	0.636	0.843	0.682	0.677	0.787
Proposed CNN	0.830	0.956	0.601	0.738	0.795	0.945	0.634	0.759	0.870	0.719	0.803	0.835
Training = 60%												
Model	**Dataset (a)**				**Dataset (b)**				**Dataset (c)**			
	AR	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**
LR	0.851	0.614	0.519	0.562	0.654	0.586	0.531	0.557	0.898	0.748	0.888	0.812
SVM	0.700	0.917	0.027	0.052	0.676	0.727	0.094	0.167	0.810	0.838	0.290	0.431
DNN	0.851	0.618	0.429	0.497	0.732	0.375	0.562	0.450	0.863	0.683	0.600	0.635
1D-CNN	0.890	0.730	0.500	0.594	0.719	0.944	0.531	0.680	0.812	0.610	0.615	0.744
Proposed CNN	0.846	0.931	0.669	0.779	0.834	0.952	0.720	0.819	0.919	0.825	0.876	0.897
Training = 70%												
Model	**Dataset (a)**				**Dataset (b)**				**Dataset (c)**			
	AR	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**
LR	0.852	0.633	0.496	0.556	0.678	0.714	0.400	0.513	0.907	0.725	0.930	0.815
SVM	0.700	0.833	0.030	0.058	0.660	0.500	0.094	0.158	0.833	0.793	0.324	0.460
DNN	0.855	0.636	0.411	0.494	0.833	0.692	0.360	0.474	0.856	0.660	0.624	0.636
1D-CNN	0.875	0.778	0.398	0.527	0.735	0.833	0.600	0.698	0.840	0.750	0.500	0.750
Proposed CNN	0.893	0.839	0.690	0.757	0.844	0.952	0.756	0.850	0.920	0.785	0.966	0.904
Training = 80%												
Model	**Dataset (a)**				**Dataset (b)**				**Dataset (c)**			
	AR	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**	**AR**	**PR**	**RR**	**F1**
LR	0.837	0.610	0.472	0.532	0.652	0.529	0.529	0.529	0.917	0.712	0.977	0.824
SVM	0.695	0.778	0.035	0.066	0.660	0.615	0.094	0.163	0.833	0.684	0.302	0.419
DNN	0.856	0.630	0.434	0.511	0.784	0.522	0.706	0.600	0.857	0.653	0.630	0.635
1D-CNN	0.859	0.800	0.359	0.496	0.714	0.909	0.588	0.741	0.833	0.705	0.574	0.762
Proposed CNN	0.723	0.908	0.742	0.816	0.901	0.958	0.841	0.896	0.958	0.857	1.000	0.947

It is also worth noting that the proposed model had better universality and performance in the realistic dataset. Comparing dataset (c) with the realistic dataset (a) and dataset (b), the F1 of the proposed model with dataset (c) reached about 0.95, but 0.816 for dataset (a) and 0.896 for dataset (b) when the training ratio was 80%. This is mainly because that the electricity theft data in dataset (c) were artificially generated, of which the data features can be identified and extracted easily by machine learning models. However, realistic electricity theft data are more complicated and lack regularity. Therefore, the results in realistic datasets are relatively worse than those in dataset (c). However, compared with other models, the performance of the proposed model was still the highest.

The comparing results show that the proposed model has better performance overall, which implies that the proposed model has higher accuracy in electricity theft detection.

5.2. Parameter Study

To study the effect of the length of the intercepted window T on the proposed model, we conducted an experiment on dataset (a) and dataset (b) by changing the value of T from 10 to 500 with a step size of 10. Figure 8a,b shows the experiment results of dataset (a) and dataset (b), respectively.

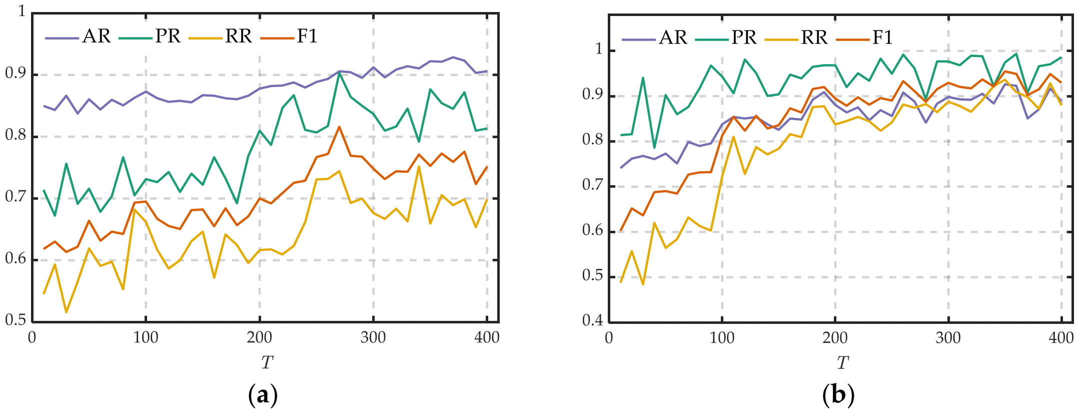


Figure 8. Performances of different T. (**a**) Trends of indicators in dataset (a); (**b**) trends of indicators in dataset (b).

The length of the intercepted window has an important impact on the performance, as shown in Figure 8. Four indicators all increased with the increasing of T at first. Then they no longer increased and began to fluctuate when T exceeded a certain value and continued to increase. In Figure 8a, PR, RR and F1 achieved their maxima when T was about 270. In Figure 8b, four indicators achieved their maxima when T was about 200. This is mainly because that the features of electricity theft data are easier to be extracted with more electricity consumption information when T increases, which leads to the improvement of the performance.

Therefore, to achieve the best performance of the proposed model, it is necessary to investigate an appropriate length of the intercepted window T for electricity theft detection.

5.3. Data Augmentation Analysis

The proposed data augmentation method was used to augment the electricity theft data in dataset (b). To study the effectiveness of the proposed data augmentation method, we varied the value of AG, which represents the repeated times, from 0 to 20 with a step of 1. At the same time, other parameters were fixed.

The comparison results of different AG in training ratios of 50% and 80% are given in Figure 9a,b respectively. The four indicators all increased at first as AG increased, while the classification accuracy decreased after AG exceeded a value. The indicators had a positive relationship with AG in the early stage because the increase in the amount of electricity theft data during the training was of great help to the classification accuracy, which can effectively increase the number of TP (true positives) in the classification result. Therefore, all four indicators had an upward trend.

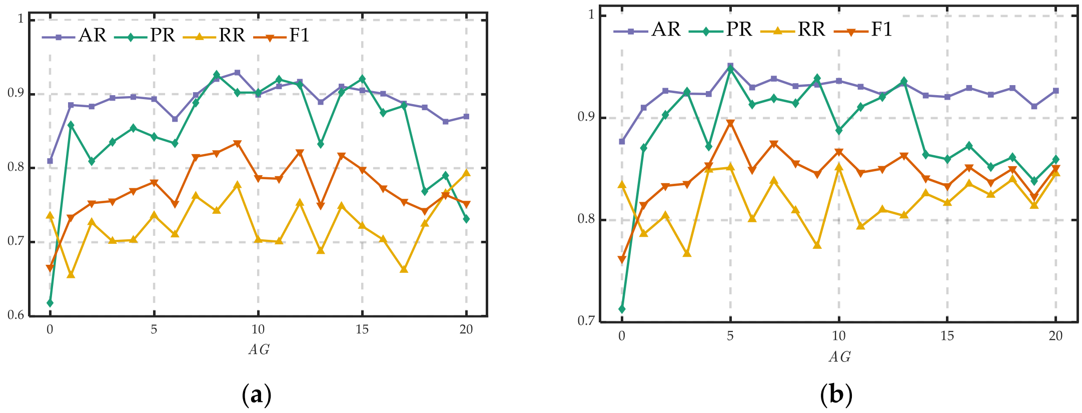


Figure 9. Performances of different AG. (**a**) Trends of indicators with a 50% training ratio; (**b**) trends of indicators with an 80% training ratio.

When *AG* continued to increase, AR, RR and F1 dropped, while PR fluctuated. The main reason is that the normal data were labeled as electricity theft during the data preprocessing when the repeated time was too large, so the training model tended to classify the normal users into abnormal users. As a result, the numbers of FP (false positives) and FN (false negatives) increased, while the numbers of TP and TN (true negative) declined in the classification result. Therefore, the classification accuracy dropped and most indicators decreased, especially the most important indicator F1.

All in all, the data augmentation which increases the number of electricity theft data points for CNN training can improve the classification accuracy effectively. It is also important to choose an appropriate *AG* to achieve better classification results, because the indicators for accuracy may fluctuate with inappropriate *AG*.

6. Conclusions

In this paper, we propose a novel electricity theft detection scheme based on TextCNN. We innovatively formulated the electricity data into two-dimensional time-series in order to capture the intraday and daily correlations of electricity consumption data. Then, we discussed the relationship between DNN, CNN and TextCNN, and explained why TextCNN is the most suitable classifier for our purposes, considering both the efficiency and effectiveness. Additionally, in order to balance the electricity consumption dataset, we proposed a data augmentation method. We conducted extensive experiments on different realistic datasets to prove the effectiveness of the proposed scheme, including the residential and industrial datasets from a province in China and the public Irish residential dataset. The experimental results show that the proposed method outperforms other methods, such as LR, SVM, DNN and 1D CNN. At the same time, we analyzed the importance and effectiveness of data augmentation.

Author Contributions: Conceptualization and methodology, X.F. and H.H.; software, Z.L. and Y.Y.; validation, W.G., H.Q. and H.F.; writing, H.H., Z.L., Y.Y. and C.Y.; supervision and project administration, X.F., H.Q., H.F. and Y.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Project Supported by the China Southern Power Grid Corporation, grant number GDKJXM20185800.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Depuru, S.S.S.R.; Wang, L.; Devabhaktuni, V. Electricity theft: Overview, issues, prevention and a smart meter based approach to control theft. *Energy Policy* **2011**, *39*, 1007–1015. [CrossRef]
2. Venkatachary, S.K.; Prasad, J.; Samikannu, R. Overview, issues and prevention of energy theft in smart grids and virtual power plants in Indian context. *Energy Policy* **2017**, *110*, 365–374. [CrossRef]
3. Northeast Group, LLC. Electricity Theft & Non-Technical Losses: Global Markets, Solutions, and Vendors. Available online: http://www.northeast-group.com/reports/Brochure-Electricity%20Theft%20&%20Non-Technical%20Losses%20-%20Northeast%20Group.pdf (accessed on 20 September 2020).
4. Liu, Z. Over 110 MWh in 35 Years, Electricity Theft Arrested in Shaoyang. Available online: http://epaper.voc.com.cn/sxdsb/html/2018-08/02/content_1329743.htm?div=-1 (accessed on 20 September 2020).
5. Messinis, G.M.; Hatziargyriou, N.D. Review of non-technical loss detection methods. *Electr. Power Syst. Res.* **2018**, *158*, 250–266. [CrossRef]
6. Short, T.A. Advanced Metering for Phase Identification, Transformer Identification, and Secondary Modeling. *IEEE Trans. Smart Grid* **2013**, *4*, 651–658. [CrossRef]
7. Leite, J.B.; Mantovani, J.R.S. Detecting and Locating Non-Technical Losses in Modern Distribution Networks. *IEEE Trans. Smart Grid* **2018**, *9*, 1023–1032. [CrossRef]
8. Jiang, R.; Lu, R.; Wang, Y.; Luo, J.; Shen, C.; Shen, X. Energy-theft detection issues for advanced metering infrastructure in smart grid. *Tsinghua Sci. Technol.* **2014**, *19*, 105–120. [CrossRef]

9. Glauner, P.; Dahringer, N.; Puhachov, O.; Meira, J.A.; Valtchev, P.; State, R.; Duarte, D. Identifying Irregular Power Usage by Turning Predictions into Holographic Spatial Visualizations. In Proceedings of the 2017 IEEE International Conference on Data Mining Workshops (ICDMW), New Orleans, LA, USA, 18–21 November 2017; pp. 258–265.
10. Buzau, M.-M.; Tejedor-Aguilera, J.; Cruz-Romero, P.; Gomez-Exposito, A. Hybrid Deep Neural Networks for Detection of Non-Technical Losses in Electricity Smart Meters. *IEEE Trans. Power Syst.* **2020**, *35*, 1254–1263. [CrossRef]
11. Jokar, P.; Arianpoo, N.; Leung, V.C.M. Electricity Theft Detection in AMI Using Customers' Consumption Patterns. *IEEE Trans. Smart Grid* **2016**, *7*, 216–226. [CrossRef]
12. Nagi, J.; Yap, K.S.; Tiong, S.K.; Ahmed, S.K.; Mohamad, M. Nontechnical Loss Detection for Metered Customers in Power Utility Using Support Vector Machines. *IEEE Trans. Power Deliv.* **2010**, *25*, 1162–1171. [CrossRef]
13. Nagi, J.; Yap, K.S.; Tiong, S.K.; Ahmed, S.K.; Nagi, F. Improving SVM-Based Nontechnical Loss Detection in Power Utility Using the Fuzzy Inference System. *IEEE Trans. Power Deliv.* **2011**, *26*, 1284–1285. [CrossRef]
14. Jindal, A.; Dua, A.; Kaur, K.; Singh, M.; Kumar, N.; Mishra, S. Decision Tree and SVM-Based Data Analytics for Theft Detection in Smart Grid. *IEEE Trans. Ind. Inf.* **2016**, *12*, 1005–1016. [CrossRef]
15. Wu, R.; Wang, L.; Hu, T. AdaBoost-SVM for Electrical Theft Detection and GRNN for Stealing Time Periods Identification. In Proceedings of the IECON 2018—44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, 21–23 October 2018; pp. 3073–3078.
16. Adil, M.; Javaid, N.; Qasim, U.; Ullah, I.; Shafiq, M.; Choi, J.-G. LSTM and Bat-Based RUSBoost Approach for Electricity Theft Detection. *Appl. Sci.* **2020**, *10*, 4378. [CrossRef]
17. Zheng, Z.; Yang, Y.; Niu, X.; Dai, H.-N.; Zhou, Y. Wide and Deep Convolutional Neural Networks for Electricity-Theft Detection to Secure Smart Grids. *IEEE Trans. Ind. Inf.* **2018**, *14*, 1606–1615. [CrossRef]
18. Hasan, M.N.; Toma, R.N.; Nahid, A.-A.; Islam, M.M.M.; Kim, J.-M. Electricity Theft Detection in Smart Grid Systems: A CNN-LSTM Based Approach. *Energies* **2019**, *12*, 3310. [CrossRef]
19. Kim, T.T.; Poor, H.V. Strategic Protection Against Data Injection Attacks on Power Grids. *IEEE Trans. Smart Grid* **2011**, *2*, 326–333. [CrossRef]
20. Zanetti, M.; Jamhour, E.; Pellenz, M.; Penna, M.; Zambenedetti, V.; Chueiri, I. A Tunable Fraud Detection System for Advanced Metering Infrastructure Using Short-Lived Patterns. *IEEE Trans. Smart Grid* **2019**, *10*, 830–840. [CrossRef]
21. Wang, X. Analysis of Typical Electricity Theft Cases—Adjust the Metering Time of Meters to Avoid the Peak Period Tariffs. Available online: https://www.zhangqiaokeyan.com/academic-conference-cn_meeting-7953_thesis/020222030513.html (accessed on 22 September 2020).
22. Han, W.; Xiao, Y. Combating TNTL: Non-Technical Loss Fraud Targeting Time-Based Pricing in Smart Grid. In Proceedings of the Cloud Computing and Security, Nanjing, China, 29–31 July 2016; Volume 10040, pp. 48–57.
23. Dhillon, A.; Verma, G.K. Convolutional neural network: A review of models, methodologies and applications to object detection. *Prog. Artif. Intell.* **2020**, *9*, 85–112. [CrossRef]
24. Jiao, J.; Zhao, M.; Lin, J.; Liang, K. A comprehensive review on convolutional neural network in machine fault diagnosis. *Neurocomputing* **2020**, *417*, 36–63. [CrossRef]
25. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
26. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
27. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1746–1751.
28. Zhang, Y.; Wallace, B. A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification. In Proceedings of the Eighth International Joint Conference on Natural Language Processing, Taipei, Taiwan, 27 November–1 December 2017; Volume 1, pp. 253–263.

29. Szegedy, C.; Wei, L.; Yangqing, J.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9.
30. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
31. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015.
32. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
33. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection: A survey. *ACM Comput. Surv.* **2009**, *41*, 15:1–15:58. [CrossRef]
34. Haixiang, G.; Yijing, L.; Shang, J.; Mingyun, G.; Yuanyue, H.; Bing, G. Learning from class-imbalanced data: Review of methods and applications. *Expert Syst. Appl.* **2017**, *73*, 220–239. [CrossRef]
35. Zhou, L. Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods. *Knowl. Based Syst.* **2013**, *41*, 16–25. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.