Hindawi
Complexity
Volume 2020, Article ID 9084704, 17 pages
https://doi.org/10.1155/2020/9084704

WILEY | Hindawi

Research Article

Effect Improved for High-Dimensional and Unbalanced Data Anomaly Detection Model Based on KNN-SMOTE-LSTM

Fuguang Bao,[1,2,3,4] **Yongqiang Wu**,[2] **Zhaogang Li**,[2] **Yongzhao Li**,[2,3] **Lili Liu**,[2] **and Guanyu Chen**[4]

[1]*Contemporary Business and Trade Research Center, Zhejiang Gongshang University, Hangzhou 310018, China*
[2]*Zhejiang Wellsun Intelligent Technology Co.,Ltd., Hangzhou 310018, China*
[3]*School of Telecommunication Engineering, Xidian University, Xian 710126, China*
[4]*School of Management Science & Engineering, Zhejiang Gongshang University, Hangzhou 310018, China*

Correspondence should be addressed to Fuguang Bao; baofuguang@126.com

Received 10 June 2020; Revised 1 September 2020; Accepted 7 September 2020; Published 17 September 2020

Academic Editor: Roberto Natella

High-dimensional and unbalanced data anomaly detection is common. Effective anomaly detection is essential for problem or disaster early warning and maintaining system reliability. A significant research issue related to the data analysis of the sensor is the detection of anomalies. The anomaly detection is essentially an unbalanced sequence binary classification. The data of this type contains characteristics of large scale, high complex computation, unbalanced data distribution, and sequence relationship among data. This paper uses long short-term memory networks (LSTMs) combined with historical sequence data; also, it integrates the synthetic minority oversampling technique (SMOTE) algorithm and K-nearest neighbors (kNN), and it designs and constructs an anomaly detection network model based on kNN-SMOTE-LSTM in accordance with the data characteristic of being unbalanced. This model can continuously filter out and securely generate samples to improve the performance of the model through kNN discriminant classifier and avoid the blindness and limitations of the SMOTE algorithm in generating new samples. The experiments demonstrated that the structured kNN-SMOTE-LSTM model can significantly improve the performance of the unbalanced sequence binary classification.

1. Introduction

With the continuous growth of urban population and wealth accumulation, the urban security has been shown in evidence, while it also faces more and more security challenges. It puts forward more austere requirements on the ability of urban disaster prevention and emergency response as various social accidents increase and the terrorism threats happen. Such technologies as big data, the Internet of things, cloud computing, and artificial intelligence have been continuously applied in the field of intelligent monitoring, which have exerted a far-reaching impact on the smart firefighting and provided more safety and security for our daily life as well as improved the efficiency and value of urban management. Monitoring methods are gradually shifting from manual audit to model prediction and automated decision-making.

Machine learning has proven to be effective in many fields, and in the context of wireless sensor network (WSN), it has proven adequate to detect attacks. High-dimensional and unbalanced data anomaly detection is common, such as detection of electricity theft behavior, anomaly detection in wireless sensor network, and anomaly detection in IDS detection. Effective anomaly detection is essential for problem or disaster early warning and maintaining system reliability. In this work, our main objective is to improve the effect of high-dimensional and unbalanced data anomaly detection. And, anomaly detection in wireless sensor network is a typical problem of high dimensional and unbalanced data anomaly detection [1].

Heterogeneous wireless sensor network is a source to represent such massive different information as light, temperature, and humidity. An important research issue related to the analysis of the sensor data is the detection of anomalies. Anomaly detection systems are usually based on the expert analysis method and data analysis method or a combination of them [2, 3]. The expert analysis method tries to find out the specific abnormal scenes through rules. The accuracy of this method has huge connection with the knowledge of experts, which is subjective, and the results are unscientific and poor in interpretability. The data analysis method is based on machine learning algorithms, which can improve the performance of the system through learning the characteristics of anomalous data. The model will provide the corresponding judgment under new situation. Common machine learning algorithms include logistic regression, support vector machine (SVM), and gradient boosted decision tree (GBDT).

Wireless sensor network (WSN) is a distributed network architecture consisting of a set of autonomously networked electronic devices (sensor nodes) that collect data from the surrounding environment. Such data as current, voltage, power, temperature, humidity, light, and noise will be collected. The market of wireless sensor network is ever growing just that advances in technology and computing [4]. Meanwhile, it is necessary to use effective network management technology to cope with the complexity of network and the large amount and variety of sensor data [5, 6]. WSN usually connects to the cloud services via the Internet. Cloud platform provides the storage and computing infrastructures which are necessary for filing and processing the huge amount of data produced by sensors [7].

A challenging study in this paper is the sensor data analysis for automatic anomaly detection [8]. In this paper, we focused on detecting abnormal changes in the sensing data which may be caused by the sensor system itself or the environment under scrutiny. For wireless sensor network, the causes of anomalies may be related to the following factors: the devices running out of power, the devices deviating from the expected behavior, and the device failing. However, it is difficult to tell an anomaly in a sensor system from a real anomaly of the sensed environment. In this case, the type of wireless sensor network, the detection method, and the interested type of exceptions may trigger a significant impact on the solution design.

This study covers following innovative points as it contains a basic classifier based on LSTM network being the anomaly detection and the structured modeling integration constructing the WSN anomaly detection system.

(1) Considering that the distribution of the wireless sensor data will change with time, this study adopts the LSTM-based anomaly detection network model to classify data, which can effectively process time-domain sequence data.

(2) The data distribution of wireless sensor data is unbalanced [9], which means that abnormal data cover a small portion of all daily monitoring data. We use the SMOTE algorithm to amplify the data to solve the overfitting caused by unbalanced data.

(3) As the SMOTE algorithm would produce noise data, influencing the determination of classification boundary, we adopt the discriminant classifier based on kNN algorithm and the basic classifier based on LSTM to screen out the valid samples and remove the noise samples, which can effectively improve the performance and accuracy of classification.

(4) The experiment shows that the defects of misclassification of the traditional method can be solved through the model based on basic classifier LSTM, data generator, and discriminant classifier and circulated organic structural fusion can be achieved.

2. Analysis of High-Dimensional and Unbalanced Data Anomaly Detection

2.1. Modeling of High-Dimensional and Unbalanced Data Anomaly Detection. In the wireless sensor network anomaly detection, Rassam et al. [10] presented the challenges of anomaly detection in WSN and put forward requirements for designing efficient and effective anomaly detection models. Abduvaliyev et al. [11] designed an intrusion detection system for the vital areas in wireless sensor networks. Steiger et al. [12] analysed time-series similarities for anomaly detection in sensor networks. Peña [13] designed a rule-based system to detect energy efficiency anomalies in smart buildings.

In the data classification and learning of high-dimensional and unbalanced data, the problem of unbalanced and multiview data classification remains unexplored in the field of network neuroscience. An intuitive approach is to obtain a balanced distribution of data through sampling methods, which can be oversampling, undersampling, or synthetic sampling. One advanced synthetic sampling method called SMOTE [14] augments artificial examples created by interpolating neighboring data points. Following this work, safe-level SMOTE [15] proposes a weighted generation process to make the data synthetic process more robust. The hybrid strategy is always chosen, which combines multiple techniques from one or both categories. Sun et al. [16] investigated the combination of the time weighting resampling method and Adaboost ensembling. The diversified ensemble learning framework, which finds the best classification algorithm for each individual subdataset, is proposed in the literature [17, 18]. Graa and Rekik [19] proposed a multiview learning-based data proliferator (MV-LEAP) that enables the classification of imbalanced multiview representations. Shi et al. [20] proposed a general multiple distribution selection method for imbalanced data classification, by proving that traditional classification methods that use single softmax distribution are limited for modeling complex and imbalanced data. Fu et al. [21] proposed an algorithm called sssHD to achieve stable sparse feature selection applied it to complicated class-imbalanced data.

In the machine learning and data analysis of high-dimensional data anomaly detection, Bereziński et al. [22] proved that an entropy-based approach is suitable to detect modern botnet-like malware based on anomalous patterns

in network. Flores et al. [23] presented a continuous hidden markov model for network anomaly detection. Huang et al. [24] proposed a natural outlier (NOF) to measure outliers. Robinson and Aria [25] used hidden Markov model (HMM) to verify the data validity of the method. Khan et al. [26] proposed a scalable and hybrid IDS based on Spark ML and the convolutional-LSTM (Conv-LSTM) network. Ergen and Kozat [27] studied unsupervised anomaly detection with LSTM neural networks. Dainius and Goranin [28] analysed the applicability of such complex dual-flow DL methods as long short-term memory fully convolutional network (LSTM-FCN) and gated recurrent unit (GRU)-FCN for the task specified on the attack-caused Windows OS system call trace dataset (AWSCTD) and compared it with vanilla single-flow convolutional neural network (CNN) models.

Compared with these methods, our proposed model can be applied not only to traditional machine learning models but also to the training of neural networks. Our model can be combined with sampling methods and can be incorporated with various hybrid strategies.

Wireless sensor network anomaly detection is essentially an unbalanced sequence binary classification. It can be said that wireless sensor anomaly detection is a challenging machine learning issue [29]. Data for such issue have three characteristics:

(1) The distribution of data will change with time

(2) The distribution of data is unbalanced, and abnormal data covers a small portion of all monitoring data.

(3) Anomaly detection is essentially a continuous unbalanced classification task

In the construction of a wireless sensor network anomaly detection system, feature selection is important for accurate classification. In wireless sensor data sets, the feature property set is similar. The feature properties are from sound-light alarm data, alarm data, electricity equipment data (daily freezing/real-time), fault data, gas data, smoke data, intelligent terminal data, water pressure data, water level data, etc.

Definition 1. Sample feature set X_i refers to the data of wireless sensor network, that is, the i^{th} sample feature set is $X_i = \{x_{i1}, x_{i2}, \ldots, x_{im}\}$, $i = 1, 2, \ldots, n$, which means there are n samples and m characteristics, x_{im} means the m^{th} feature of the ith sample feature in data set X_i.

Definition 2. Sample classification Y_i refers to whether the sample data belongs to abnormal data of wireless sensor network. 1 indicates abnormal data and 0 indicates normal data. The $Y_i = \{y_i\}$, $y_i = \{1, 0\}$.

Definition 3. Set the historical data as V and the real-time data at time t as v_t. Then, the anomaly detection of wireless sensor network is to determine whether v_t is abnormal according to V, that is,

$$V = \begin{Bmatrix} x_{11}, & x_{12}, & \ldots & x_{1m}, & y_1 \\ x_{21}, & x_{22}, & \ldots & x_{2m}, & y_2 \\ \ldots & & \ldots & & \ldots \\ x_{n1}, & x_{n2}, & \ldots & x_{nm}, & y_n \end{Bmatrix}, \quad (1)$$
$$v_t = \{x_{t1}, x_{t2}, \ldots, x_{tm}\}.$$

Thus, anomaly detection in wireless sensor network is an unbalanced dichotomy issue, and the data sample size is large, the computational complexity is high, the data distribution is unbalanced, and there will be a sequence relationship among data.

2.2. LSTM Modeling. Recurrent neural network (RNN) is a deep learning model for processing time series data [30]. Its special network structure can make the neuron output act directly on itself as the input in next moment. The neural network output is the result of the interaction between the input of the moment and the states of all moments, aiming to achieve the purpose of sequence modeling. It can learn characteristics and long-term dependencies from sequence and time series data. The recurrent neural network with sigmoid activation function has been proved to be Turing-complete by Schafer and Zimmermann in 2006, which means that RNN can perform the same calculation as any computable program, given the correct weight [31]. However, the gradient at the current moment can only deliver a finite layer to the historical moment, indicating that RNN cannot solve the problem of long-term dependence.

In 1997, Hochreiter and Schmidhuber proposed the long short-term memory networks (LSTMs) based on RNN and introduced CEC (constant error carousel) unit to solve the gradient explosion and disappearance problem of BPTT (backpropagation time) [32]. In 2001, Felix Gers further improved the network structure of LSTM by adding Forgotten Gate and Peephole Connection [33]. In 2012, Alex Graves proposed the CTC (connectionist temporal classification) training criterion of LSTM [34]. In 2014, Chung proposed the GRU (gate recurrent unit), which integrated the input gate and forgotten gate of LSTM into an updated gate to reduce the parameters and train faster [35].

The LSTM model preserves a long-term memory by such unique gates as forgotten gate, input gate, output gate, and the memory unit, which is denoted as C_t. Figure 1 presents the schematic diagram of LSTM cell structure.

The state of LSTM depends on the current input and the previous state [37], while the latter in turn depends on the even more previous input and state. Suppose that the hidden unit uses the sigmiod function for each time step from $t = 1$ to $t = r$ and the following updated equation is applied:

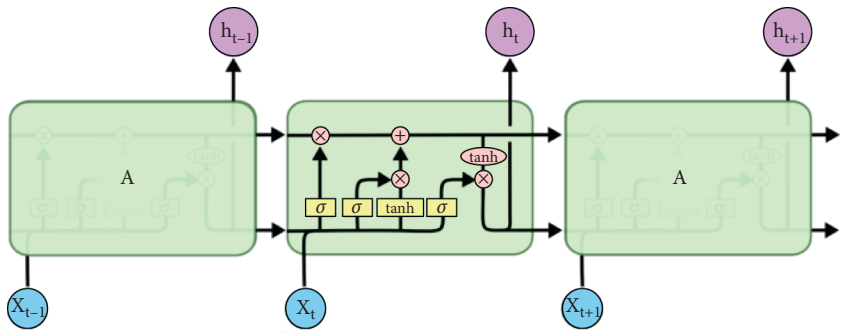

Figure 1: Schematic diagram of the LSTM cell structure [36].

$$\begin{cases} C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t, \\ f_t = \sigma\left(W_f * \left[h_{t-1}, x_t\right] + b_f\right), \\ i_t = \sigma\left(W_i * \left[h_{t-1}, x_t\right] + b_i\right), \\ \widetilde{C}_t = \tanh\left(W_c * \left[h_{t-1}, x_t\right] + b_c\right), \\ h_t = o_t * \tanh\left(C_t\right), \\ o_t = \sigma\left(W_o * \left[h_{t-1}, x_t\right] + b_o\right). \end{cases} \tag{2}$$

In the updated equation, f_t and i_t represent the forgotten gate and the input gate, respectively, which are obtained by the linear transformation of the input x_t and the output h_{t-1} of the hidden layer in the previous step and then by the activation function σ. h_t represents the output of the current hidden layer. In every moment, the forgotten gate controls the forgotten degree of the previous memory, and the input gate controls the memory degree of the new memory unit $\widetilde{C}_t$ into long-term memory. Therefore, it can be concluded that the transition from the state C_{t-1} of the previous memory unit to the current state C_t is not only entirely dependent on the state calculated by the activation function but also controlled by the forgotten gate and the input gate. o_t stands for the output gate, which refers to the controlling of how the short-term memory is affected by long-term memory. W represents the weight matrix of each neuron node in the neural network. W_i is the weight of input gate i_t, W_f is the weight of forgotten gate f_t. b is the bias of the neuron. b_f and b_i represent the bias for the forgotten gate and the input gate, respectively.

In a well-trained LSTM model, when there is no important information in the input sequence, the value of the forgotten gate of LSTM is close to 1 and the value of the input gate is close to 0, while the past memory will be saved, thus realizing the long-term memory function; when important information appears in the input sequence, this information indicates that the previous memory is no longer important, the value of the input gate is close to 1, and the value of the forgotten gate is close to 0 which means that the old memory is forgotten and the new important information is saved. After such network design, the entire model gets easier access to learning the long-term dependencies between sequences.

2.3. SMOTE Algorithm. Synthetic minority oversampling technique (SMOTE) algorithm is an improved scheme based on the random oversampling method [14]. It synthesizes new samples for the minority classes based on "linear interpolation."

The SMOTE algorithm adopts a subset of data from the minority classes as an example and then creates similar new synthetic examples. Then, the original data set will collect these synthetic examples. In this process, it generates a sample from the line between the minority class samples and their neighbors. The new data set can work as a training sample to train the classification model, which can effectively solve the problem of overfitting caused by simple random oversampling. Figure 2 illustrates the schematic diagram of the synthetic instances of the SMOTE algorithm.

Algorithm 1 shows the steps of the SMOTE algorithm.

Further, a new oversampling method borderline-stroke algorithm is proposed in the literature [38] based on SMOTE. The main idea is to generate new samples along the boundary of the minority samples, as shown in Figure 3. Bunkhumpornpat et al. [15] proposed a safe-level-smote algorithm to oversample a minority samples at a safer level by adjusting the safety ratio.

3. Anomaly Detection Model Based on KNN-SMOTE-LSTM

The wireless sensor network (WSN) anomaly detection model based on kNN-SMOTE-LSTM is a LSTM WSN detection network model based on SMOTE improvement, and kNN discriminant classifier can continuously screen the security generated samples to improve the performance of the model. Figure 4 presents the process of the model.

Considering that the distribution of the wireless sensor data will change with time and new abnormal situation may appear at any time, we adopt the LSTM-based anomaly detection network model to effectively cope with this kind of time-domain sequence data. The unbalanced data distribution of wireless sensor data, which means that abnormal data are only a small portion of all daily monitoring data, leads to the application of the SMOTE algorithm to amplify the data to solve the problem of overfitting caused by unbalanced data. As the SMOTE algorithm would produce noise data, influencing the determination of classification boundary, we adopt the discriminant classifier based on kNN algorithm and the basic classifier based on LSTM to screen out the valid samples and remove the noise samples, which can effectively improve the performance and accuracy of classification.

Before constructing the model based on kNN-SMOTE-LSTM, it is necessary to process the data according to the

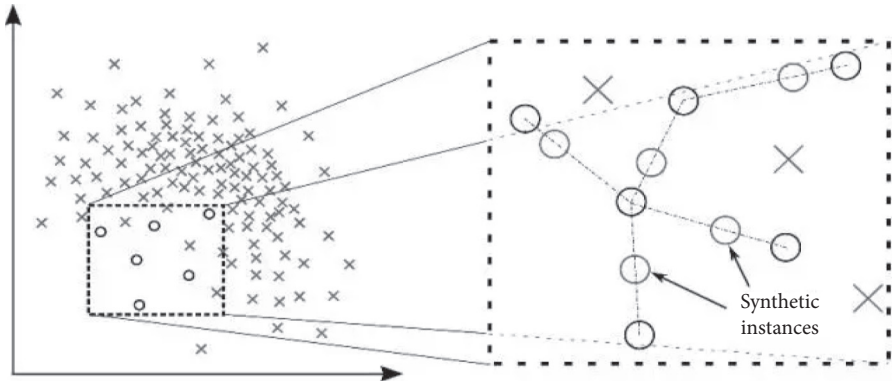

Figure 2: Schematic diagram of the synthetic instances by the SMOTE algorithm [14].

Input: T for training sample, N for oversampling rate, K for k-nearest neighbor parameter, and n for the number of the small-sized sample (minority samples $X_{\min}$)
Output: S for new training sample
Step 1: calculate the k-nearest neighbors of x_i of minority samples ($X_{\min}$) (Euclidean distance is adopted in this paper)
Step 2: randomly select a sample x_a from the k-nearest neighbors.
Step 3: generate a random number ζ between 0 and 1 for synthesis of a new sample x_{i1}
$x_{i1} = x_i + \zeta * (x_a - x_i)$.
Step 4: repeat the step 2 and step 3 according to the oversampling rate N
Step 5: obtain new training sample S
Step 6: the new training sample S was classified with a classifier
Step 7: output classification results

Algorithm 1: The SMOTE algorithm steps.

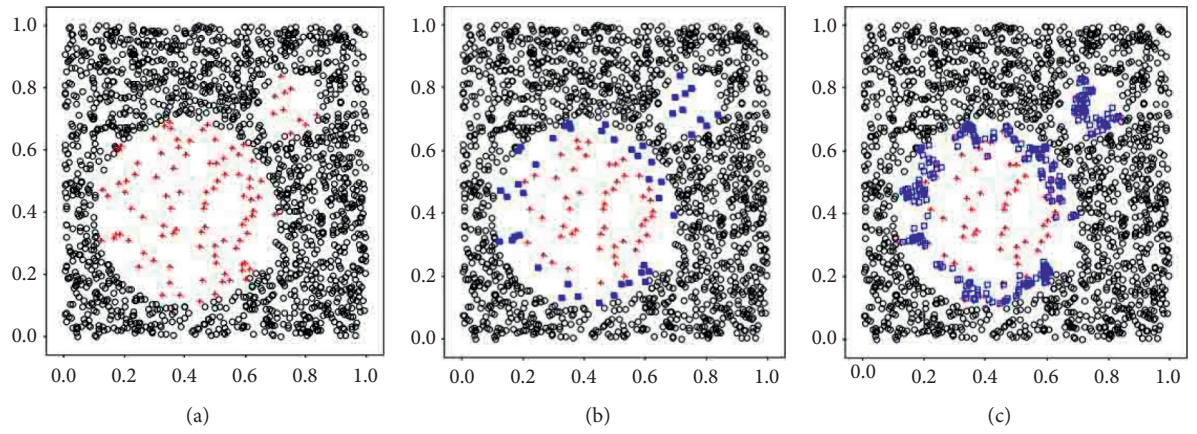

Figure 3: Schematic diagram of the synthetic instances by the borderline-SMOTE algorithm.

characteristics, removing the irrelevant features and eliminating those characteristics whose positive and negative sample distributions are particularly close, only retaining those characteristics which are highly correlated and have a large difference between positive and negative sample distributions. After that, inputting the new wireless sensor network data will see whether this situation is anomaly.

We have set several parameters as follows: X for the real data set after data preprocessing, in which the normal sample set of most classes is X_{maj} and $X_{\min}$ for the abnormal sample set of minority classes. The data generator is Z, and the data set T_t is generated by SMOTE algorithm. The discriminant classifier is D, and kNN algorithm is adopted in this paper. The basic classifier is G_t, and we adopt LSTM anomaly detection network model for the basic classifier. The algorithm steps of kNN-SMOTE-LSTM anomaly detection network model are described as follows:

(i) Step 1: kNN algorithm works to train the real data set X and construct kNN discriminant classifier D.

(ii) Step 2: by training the real data set X in formula (2) and adjusting such network parameters as the

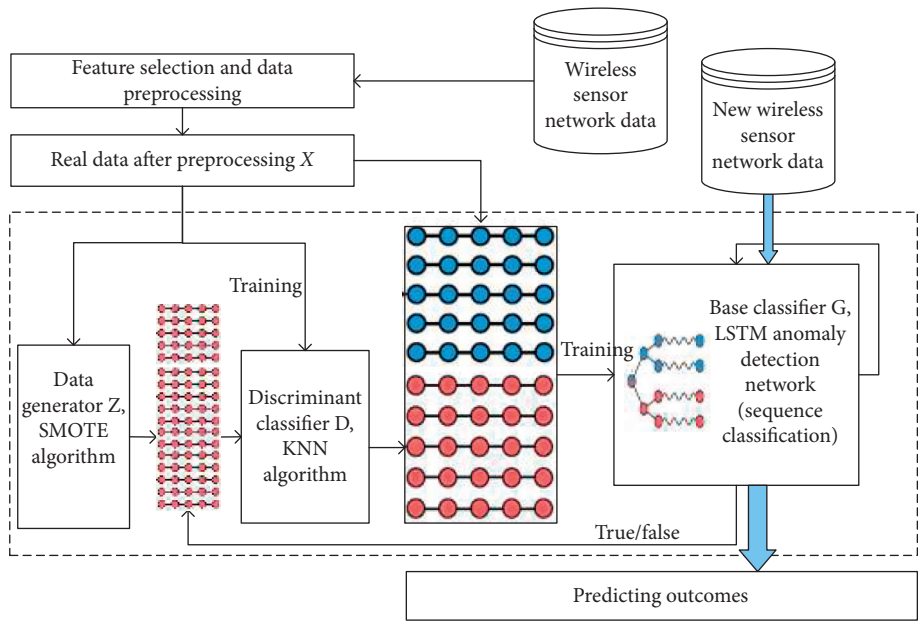

Figure 4: The process of the kNN-SMOTE-LSTM model.

number of layers and nodes, activation function, loss function, time step, learning rate, and dropout rate, the LSTM anomaly detection network basic classifier G_0 is constructed.

(iii) Step 3: calculate the k-nearest neighbors of x_i in the small-sized sample $X_{\min}$ (Euclidean distance is adopted in this paper) and randomly select a sample x_a from the k-nearest neighbors.

(iv) Step 4: generate a random number ζ between 0 and 1 for synthesis of a new sample $t_{\min 0}$:

$$t_{\min 0} = x_i + \zeta * (x_a - x_i). \tag{3}$$

(v) Step 5: the discriminant classifier D and the basic classifier G_0 work to determine whether the classification label of $t_{\min 0}$ is consistent with the classification label of $X_{\min}$. If consistent, then $t_{\min 0}$ is the valid generated sample and constitutes to be a part of the data set T_0. If not, they will be discarded.

(vi) Step 6: this is the most critical step. According to the oversampling rate N, iteration $t = 1, 2, \ldots, M$. The real data set X and the generated data set T_t constitute the training data set $X + T_t$. By generating the data set $T_1, T_2, \ldots, T_t$, the basic classifier $G_1, G_2, \ldots, G_t$ can be continuously trained. In turn, the base classifier G_t can determine whether the data in the generated data set T_t is true, that is, whether the sample is valid generated.

(vii) Step 7: after the iteration, the two types of data ($X_{\min}$, and X_{maj}) approach equilibrium, and the final classifier G_t is obtained.

(viii) Step 8: test and evaluate the test data based on G_t and output the final prediction results.

In this paper, Step 1 is the experimental step of the discriminant classifier based on kNN, Step 2 is the experimental step of the base classifier based on LSTM, and Steps 3-4 are the experimental steps of the data generator based on SMOTE. Steps 5–7 are a part of the iteration cycle of the wireless sensor network anomaly detection model based on kNN-SMOTE-LSTM. Step 8 helps to generate the final classifier for testing and evaluation.

4. Experimental Analysis and Evaluation

In this section, the real data of a wireless sensor network is mainly used for modeling and analysis. Data sources, data attributes, data preprocessing, model training, and experimental results comparison together make up the analysis. Table 1 presents the experimental environment of this paper.

4.1. Data Sources and Attributes. The data set contains real-time wireless sensor data for a scenario that occurred in August 2019, and there are 369 recorded exceptions out of 213,608 data. The data set was so lopsided that the abnormal data account for 0.173% of all records. The project aims to improve the performance of the existed anomaly detection process, improve the accuracy of anomaly detection, better interpret the anomaly patterns, and prevent anomalies through techniques under the existed data-driven strategies.

Because of confidentiality, the original characteristics and further background information about the data were not available. After the PCA (principal component analysis) extraction, it was transformed into fields of V1, V2, . . ., V28. "Time" and "Class" are the only fields out of the PCA transformation. The field "Time" represents the number of seconds between each status data and the initial status data in the data set. "Class" refers to the label of the category: 1 means the abnormal and 0 means normal. The data sample

Table 1: Experimental environment configuration.

Operating system	Processor	CPU (GHz)	Core	RAM (G)	Software
Win 10	Intel Core i5-8265U	3.4	8 cores	8	PyCharm 2017

has characteristics of large scale, complex computation, and unobservable features. Also, a better classification requires a characteristic selection through data preprocessing and the elimination of similar distribution.

4.2. Data Preprocessing. Data quality depends on a number of factors, including integrity, accuracy, consistency, timeliness, interpretability, and credibility. However, the worldwide data are easy to be affected by factors of noise, missing values, and inconsistent data. Low quality of original data will lead to low quality of data mining results. Therefore, the original data must be preprocessed.

Data preprocessing mainly includes data cleaning, data integration, data reduction, and data transformation. Effective data preprocessing can significantly improve the quality of the data and then improving the effect of the model and reducing the time consumption in the actual modeling process. The raw data are all processed structured data.

We analyze the distributions of V1, V2, . . ., V28 and use factor analysis, discrepancy analysis, and means comparison analysis to research the fields. It finds that fields of V22, V23, and V25 had serious overlap, and there is no significant difference in the independent sample T test/F test. These characteristics and fields are eliminated. Table 2 shows the difference of the measurement data by F test.

Table 2: The difference of the measurement data by F test.

Field	F test	Df	Sig.
$V1$	2955.669	1, 213608	0.000
$V2$	2393.402	1, 213608	0.000
$V3$	11014.508	1, 213608	0.000
$V4$	5163.832	1, 213608	0.000
$V5$	2592.358	1, 213608	0.000
$V6$	543.511	1, 213608	0.000
$V7$	10349.605	1, 213608	0.000
$V8$	112.548	1, 213608	0.000
$V9$	2746.6	1, 213608	0.000
$V10$	14057.98	1, 213608	0.000
$V11$	6999.355	1, 213608	0.000
$V12$	20749.822	1, 213608	0.000
$V13$	5.948	1, 213608	0.015
$V14$	28695.548	1, 213608	0.000
$V15$	5.08	1, 213608	0.024
$V16$	11443.35	1, 213608	0.000
$V17$	33979.17	1, 213608	0.000
$V18$	3584.381	1, 213608	0.000
$V19$	344.991	1, 213608	0.000
$V20$	115	1, 213608	0.000
$V21$	465.916	1, 213608	0.000
$V22$	0.185	1, 213608	0.667
$V23$	2.053	1, 213608	0.152
$V24$	14.851	1, 213608	0.000
$V25$	3.116	1, 213608	0.078
$V26$	5.654	1, 213608	0.017
$V27$	88.045	1, 213608	0.000
$V28$	25.901	1, 213608	0.000

4.3. The Training Model. In this paper, we have trained such basic classification models as Gaussian Naive Bayes (Gaussian NB), logistic regression, k-nearest neighbor classifier (kNN), BP neural network, support vector machine (SVM), AdaBoost classifier, gradient boosted decision tree (GBDT), random forest (RF), and LSTM.

We evaluate the sample data set with 10-fold cross-validation. It partitions the data set T into 10 mutual exclusive subsets T_I of similar size. Each subset T_I tries to maintain the data distribution consistency. Then, the union of 9 subsets functioned as the training data set in turn, and the remaining one subset worked as the test data set. The ultimate test evaluation result is the average of the evaluation results of the 10 tests. Obviously, ten-fold cross validation is more stable and accurate for test evaluation results.

The evaluation indexes mainly cover accuracy, precision, recall, F-score, and AUC (area under roc curve) [39, 40]. Importantly, determining a highly unbalanced data set will by default report a high baseline accuracy. For the binary classification, it is suggested that a more rigorous method should be applied like the confusion matrix which provides the true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Table 3 shows the judgment of the experimental results made by the confusion matrix which is formed according to its real category and the predicted category of classifier.

From these rates, two other key metrics could be formulated: one is precision, the fraction of moments correctly classified as abnormal sample, and the other one is recall, the fraction of abnormal moments that are correctly classified:

$$\text{precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})},$$
$$\text{recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})}. \tag{4}$$

Given the area under the precision-recall (PR) curve (AUPR), different classification algorithms are more effective than accuracy only. The P-R curve (precision-recall curve) takes the precision as the vertical axis and the recall as the horizontal axis. The ROC curve (receiver operating characteristic curve) takes the true positive rate (recall) as longitudinal axis and the false positive rate as the horizontal axis. The two curves can assess the classifier performance of classification and generalization ability. AUC refers to the areas under the ROC curve.

Table 3: Confusion matrix of the binary classification.

	Normal sample	Abnormal sample
Predicted normal	TN	FN
Predicted abnormal	FP	TP

Scikit-learn (Sklearn) is Python's algorithm library of machine learning. It achieves such common machine learning algorithms as data preprocessing, data dimensionality reduction, classification, regression, and unsupervised. In this paper, Sklearn (machine learning library of Python) is used to train Gaussian naive Bayes (GaussianNB), logistic regression, k-nearest neighbor classifier (kNN), BP neural network, support vector machine (SVM), AdaBoost classifier, gradient boosted decision tree (GBDT), and random forest (RF) [41]. Keras (deep machine learning library of Python) functions to train the LSTM model. Table 4 presents the parameters of each base classification model.

The classification report provided by scikit-learn v0.18.1 also evaluates algorithm performance. This report provides precision and recall values with respect to both classes and F-scores. The information is valuable for determining the extent to which the algorithm provides FP and FN.

Considering the randomness of the applied algorithms in this paper, it adopts the method of "fixed pseudorandom parameters + cross validation," which makes the detection results more stable and reliable and even the randomness still exist. Table 5 shows the detection results obtained as the models were trained by each base classifier. Table 5 shows that the F-score of the Gaussian naive Bayes model is the lowest with only 0.18. This is because the Gaussian naive Bayes classifier assumes conditional independence. As simple classification algorithms, logistic regression, AdaBoost classifier, and k-nearest neighbor classifier have similar evaluation results, but they are slightly inferior to BP neural network, GBDT, SVM, and other effective classification algorithms. The random forest (RF) model and the LSTM model show better comprehensive classification performance than the above classification models.

A further analysis of Table 5 shows that the random forest model and the LSTM model have the highest accuracy of 99.95%, but the accuracy is not conclusive as the test data are unbalanced. It has shown that excluding the Gaussian Naive Bayes model (this is because the assumption of conditional independence exists in the Gaussian Naive Bayes classifier; when the assumption is not established, characteristics will interact with each other, resulting in a significant reduction in the precision.), the random forest model has the highest F-score and the LSTM model has the highest AUC. By contrast, LSTM is superior to the commonly used SVM classification algorithm. The antinoise capability of the LSTM model may be better than that of SVM model when constructing the optimal hyperplane to solve the global optimal solution. The LSTM model is more suitable for this kind of unbalanced sequence classification task and can effectively cope with this kind of time-domain sequence data.

In summary, the LSTM model works as the base classifier in the anomaly detection model of wireless sensor network.

To further verify the effectiveness of the LSTM model as the basic classifier, the P-R curve and ROC curve are drawn as references, and Figure 5 pictures the detection results. Both the P-R curve and the ROC curve estimate the classification performance and generalization capability of machine learning algorithms with a given data set. Depending on Figure 5, all classification models performed well except Gaussian naive Bayes. Among the P-R curves, the random forest model and k-nearest neighbor classifier performed best, while the LSTM model had the largest area under the ROC curve, indicating that the LSTM model could be a better base classifier.

The sampling method is utilized to make up the imbalance of the data set by reducing the classes to be same in size. Undersampling and oversampling are two roughly equivalent and opposite techniques which use a bias to achieve this purpose. Such other complex algorithms as synthetic minority oversampling technique (SMOTE), and the adaptive synthetic sampling approach (ADASYN) actually create new data points based on known samples and their features rather than simply replicating the minority classes [42].

In summary, we adopt the LSTM model as the basic classifier for time-domain sequence data like wireless sensors data. For the unbalanced data distribution, it takes the SMOTE algorithm as the data generator. For the impact of noise data on the determination of the classification boundary caused by the addition of SMOTE algorithm, the discriminant classifier based on kNN and the basic classifier based on LSTM work to screen out the valid samples. Then, the kNN-SMOTE-LSTM model is constructed to carry out experiments for anomaly detection in wireless sensor network.

The parameters of kNN and LSTM are still the same as previous, and Table 6 lists the parameters of the SMOTE algorithm.

4.4. Model Verification and Experimental Results Analysis. After data preprocessing and model training of basic classifier, data generator, and discriminant classifier, the kNN-SMOTE-LSTM anomaly detection model has been constructed. Model verification is mainly to test the stability and generalization ability of the kNN-SMOTE-LSTM model.

The kNN-smote-LSTM model updates the sampling ratio in each loop iteration. The sampling ratio is the proportion of the minority samples to the majority samples. Figure 6 shows the performance of the P-R curve and ROC curve under different sampling ratios for kNN-SMOTE-LSTM. The P-R curve and the ROC curve illustrate that the results of different sampling ratios are very close. Combined with the AUC value changes shown in Figure 7, the kNN-SMOTE-LSTM model performs best when the sampling ratios are 0.7 and 1.

To further verify the effectiveness of the kNN-SMOTE-LSTM model, we compare the experimental results with that of the unbalanced oversampling algorithm which includes ADASYN, SMOTE, borderline-SMOTE, SVM SMOTE, SMOTEENN, and SMOTETomek in the combination with

Table 4: Model parameters of the basic classifier.

Model	Parameters
Gaussian naive Bayes (GaussianNB)	priors = None, var_smoothing = 1e-09
Logistic regression	penalty = "l2," dual = False, tol = 0.0001, C = 1.0, fit_intercept = True, intercept scaling = 1, class weight = None, random_state = 0, solver = "warn," max_iter = 100, multi_class = "warn," verbose = 0, warm_start = False,
AdaBoost classifier	base_estimator = none, n_estimators = 50, learning_rate = 1.0, algorithm = "SAMME.R,", random_state = 0
k-Nearest neighbor classifier (kNN)	n_neighbors = 5, weights = "uniform," algorithm = "auto," leaf_size = 30, p = 2, metric = "minkowski," metric_params = none, n_jobs = none
BP neural network	hidden_layer_sizes = (100), activation = "relu," solver = "Adam," alpha = 0.0001, batch_size = "auto," learning_rate = "constant," learning_rate_init = 0.001, power_t = 0.5, max_iter = 200, shuffle = true, random_state = 0, tol = 0.0001, verbose = false, warm_start = false, momentum = 0.9, nesterovs_momentum = True, early_stopping = false, validation_fraction = 0.1, beta_1 = 0.9, beta_2 = 0.999, epsilon = 1e-08, n_iter_no_change = 10
Gradient boosted decision tree (GBDT)	loss = "deviance," learning_rate = 0.1, n_estimators = 100, subsample = 1.0, criterion = "friedman_mse," min_samples_split = 2, min_samples_leaf = 1, min_weight_fraction_leaf = 0.0, max_depth = 3, min_impurity_decrease = 0.0, min_impurity_split = None, init = None, random_state = 0, max_features = None, verbose = 0, max_leaf_nodes = None, warm_start = False, presort = "auto," validation_fraction = 0.1, n_iter_no_change = None, tol = 0.0001
Support vector machine (SVM)	penalty = "l2," loss = "squared_hinge," dual = True, tol = 0.0001, C = 6, multi_class = "ovr," fit_intercept = True, intercept_scaling = 1, class_weight = None, verbose = 0, random_state = 0, max_iter = 1000
Random forest (RF)	n_estimators = "warn," criterion = "mse," max_depth = None, min_samples_split = 2, min_samples_leaf = 1, min_weight_fraction_leaf = 0.0, max_features = "auto," max_leaf_nodes = None, min_impurity_decrease = 0.0, min_impurity_split = None, bootstrap = True, oob_score = False, n_jobs = None, random_state = 0, verbose = 0, warm_start = False
LSTM	trainRate = 0.7, timeStep = 3, dropout = 0.5, epochs = 30, batchSize = 100 nodes = [32, 64, 16, 1], chooseAct = "relu"

Table 5: Comparison of detection results of basic classifier.

Basic classifier	Error rate	Accuracy	Precision	Recall	F-score	AUC
Gaussian naive Bayes (GaussianNB)	0.0176	0.9824	0.1026	0.8537	0.1832	0.9163
Logistic regression	0.0008	0.9992	0.8492	0.6740	0.7516	0.8296
AdaBoost classifier	0.0008	0.9992	0.8113	0.7143	0.7597	0.8611
k-nearest neighbor classifier (kNN)	0.0007	0.9993	0.9228	0.6726	0.7781	0.8373
BP neural network	0.0007	0.9993	0.8894	0.7189	0.7952	0.8591
Gradient boosted decision tree (GBDT)	0.0006	0.9994	0.9175	0.7154	0.8040	0.8604
Support vector machine (SVM)	0.0007	0.9993	0.8170	0.8022	0.8095	0.9012
Random forest (RF)	0.0005	0.9995	0.9313	0.7850	0.8519	0.8902
LSTM	0.0005	0.9995	0.8723	0.8255	0.8483	0.9132

the LSTM model [43]. Table 7 presents the detection results. The kNN-SMOTE-LSTM model showed excellent comprehensive performance.

Under the detection of real data, most of the oversampling algorithms give poor classification performance when worked without the integration of the constructed model or with only two models, and the values of F-score are lower than those of just LSTM (the F-score is 0.8483) only, including the SMOTE + LSTM model with the F-score of only 0.2276. The proposed kNN-SMOTE-LSTM model in this paper integrates the basic classifier, data generator, and discriminant classifier through the structural fusion of the model. The F-score is 0.9167, and the AUC is 0.9296 which show comprehensive classification performance and generalization ability, given that a sampling algorithm would add a lot of unnecessary noise data while leading to the deviation of final test on the training samples. However, the kNN-SMOTE-LSTM model proposed in this paper

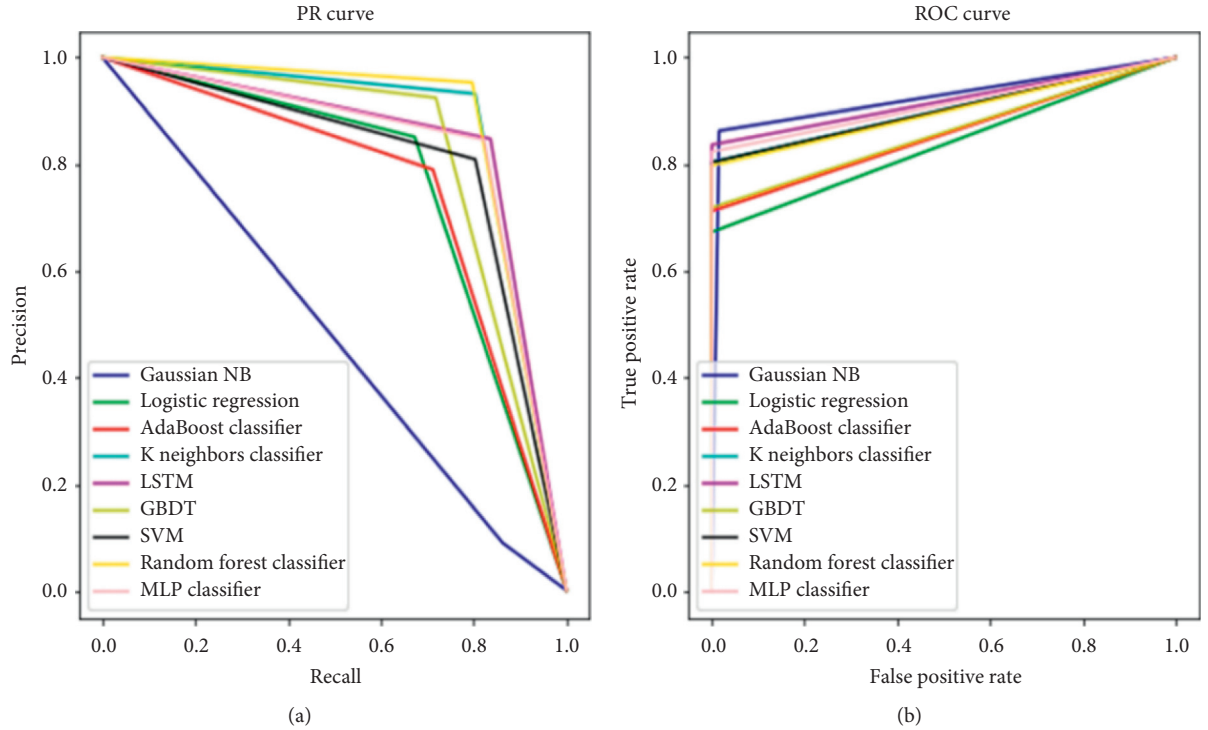

Figure 5: P-R curve and ROC curve of the basic classifiers. (a) PR curve. (b) ROC curve.

Table 6: Parameters of SMOTE.

Algorithm	Parameters
SMOTE	k_neighbors = 2, kind = "deprecated," m_neighbors = 2, n_jobs = 1, out_step = "deprecated," random_state = 0, ratio = None, sampling_strategy = "auto," svm_estimator = "deprecated"

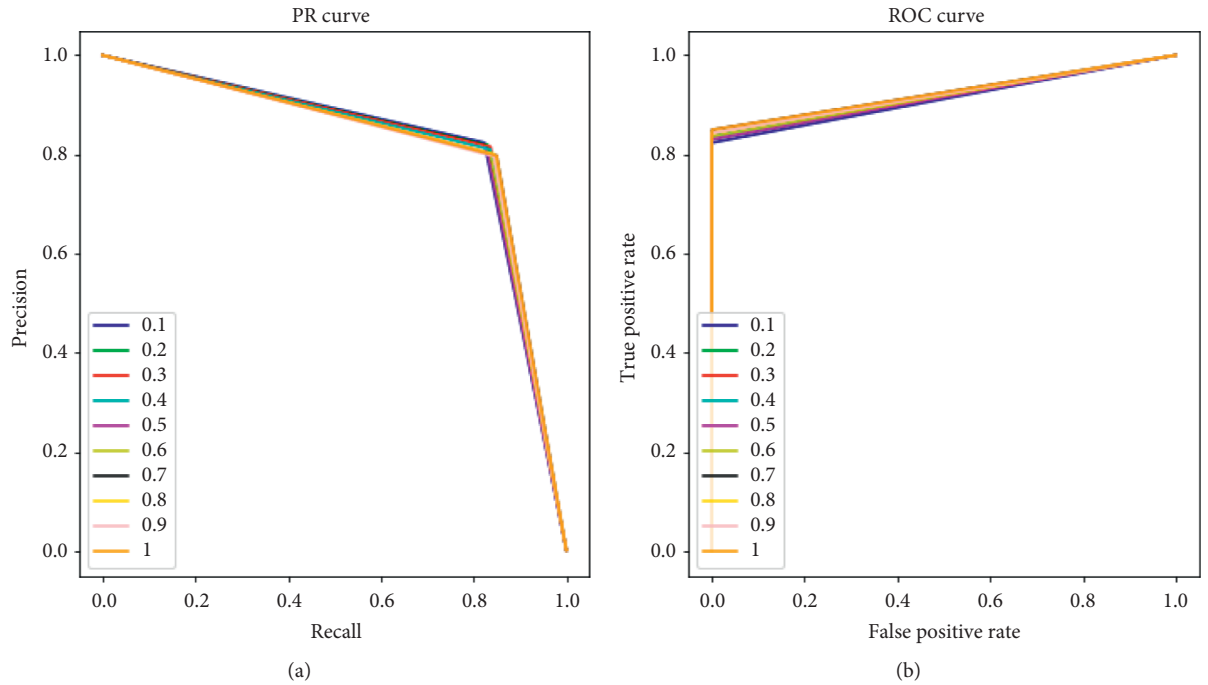

Figure 6: PR curve and ROC curve under different sampling ratios for the kNN-SMOTE-LSTM. (a) PR curve. (b) ROC curve.

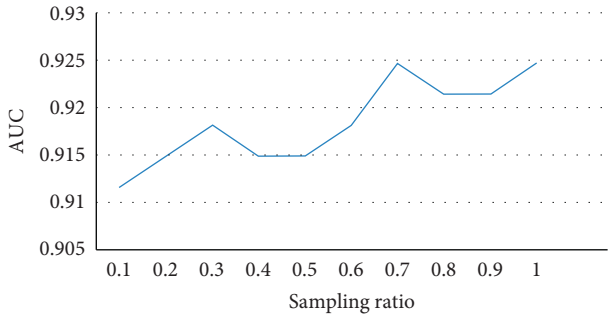

Figure 7: AUC value for kNN-SMOTE-LSTM under different sampling ratios.

Table 7: Comparison of detection results of unbalanced algorithm.

Model	Precision	Recall	F-score	AUC
LSTM	0.8723	0.8255	0.8483	0.9132
ADASYN + LSTM	0.0301	0.7718	0.0580	0.9246
SMOTE + LSTM	0.1300	0.9128	0.2276	0.9283
Borderline-SMOTE + LSTM	0.8095	0.7987	0.8041	0.9021
Svm SMOTE + LSTM	0.7669	0.8389	0.8013	0.9073
SMOTEENN + LSTM	0.1367	0.8523	0.2372	0.9275
SMOTETomek + LSTM	0.1373	0.8725	0.2356	0.9387
kNN-SMOTE-LSTM (this work)	0.9496	0.8859	0.9167	0.9296

Note. "+" indicates the combination of models and "–" indicates the structural fusion of models.

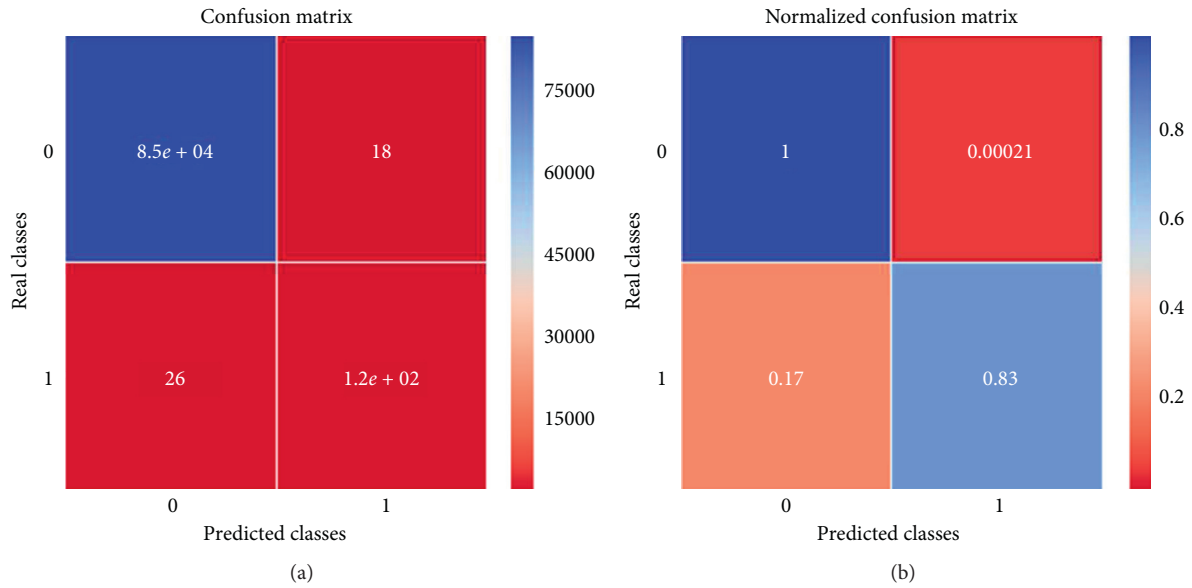

Figure 8: (a) Confusion matrix and (b) normalized confusion matrix for LSTM.

improves the classification performance through the rigorous designed structured network model and the organizational fusion of the model to continuously iterate the sampling ratio and screen the valid samples.

To verify the assumption, confusion matrices [44] of each model are shown in Figures 8–15, LSTM in Figure 8, ADASYN + LSTM in Figure 9, SMOTE + LSTM in Figure 10, Borderline-SMOTE + LSTM in Figure 11, SVM SMOTE + LSTM in Figure 12, SMOTEENN + LSTM in Figure 13, SMOTETomek + LSTM in Figure 14, and kNN-SMOTE-LSTM in Figure 15.

Experiment results show that the directly used sampling algorithm can improve the accuracy of samples of the minority class, but it will cause serious mistake for the majority class sample classification and improve the rate of FP. The FP rate of the base classifier LSTM model is

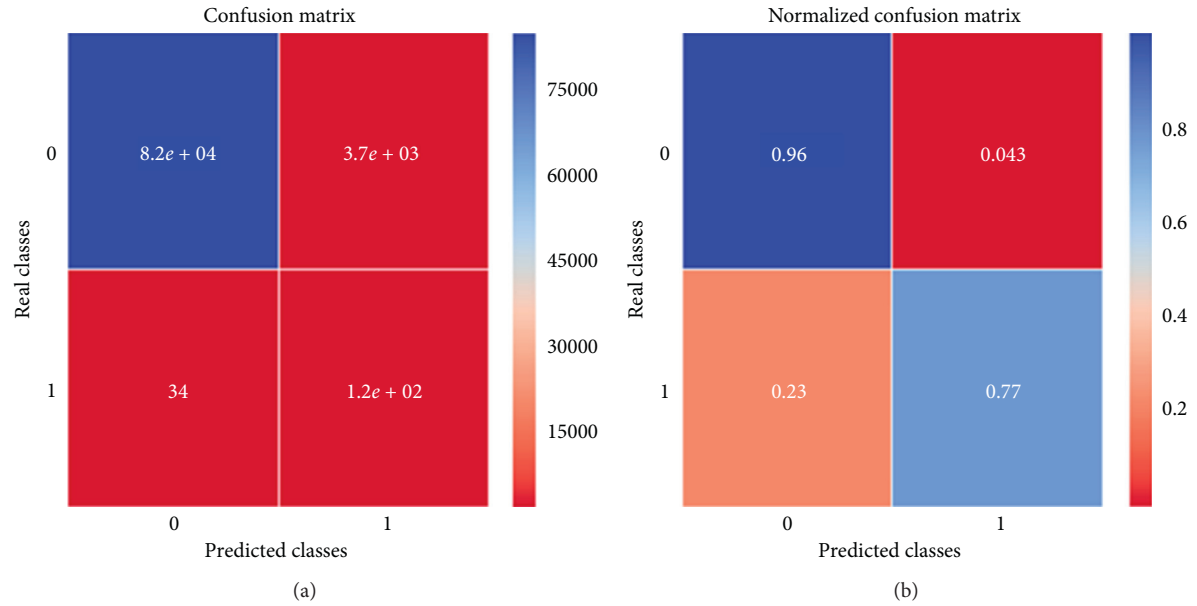

Figure 9: (a) Confusion matrix and (b) normalized confusion matrix for ADASYN + LSTM.

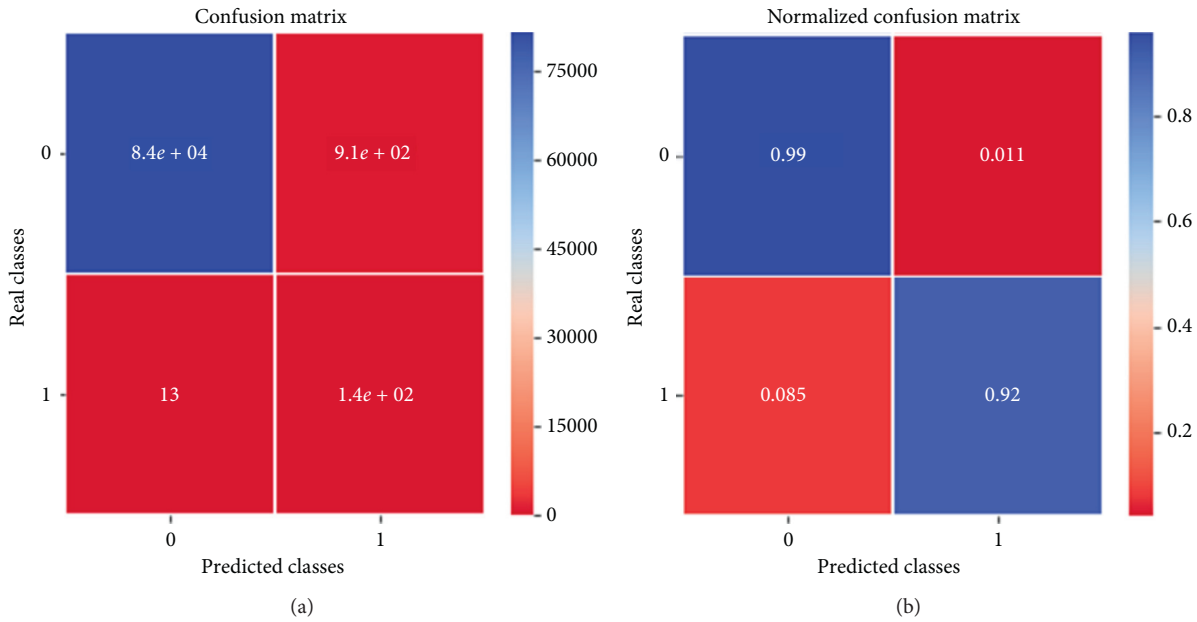

Figure 10: (a) Confusion matrix and (b) normalized confusion matrix for SMOTE + LSTM.

0.00021with the number of 18. The FP rate for ADASYN + LSTM model is 0.043 with the number of 3706, and the FP rate for SMOTE + LSTM model is 0.011. Compared with Figure 15, the kNN-SMOTE-LSTM model improved this situation. It does not only enhance the accuracy of minority samples but also solve the problem of the

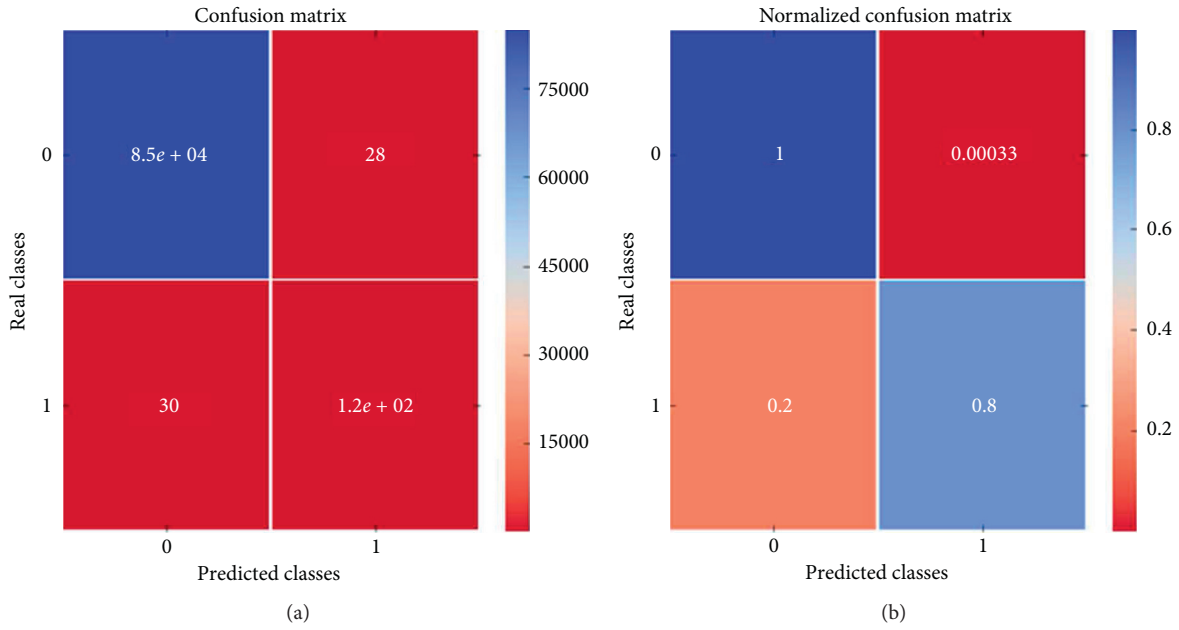

Figure 11: (a) Confusion matrix and (b) normalized confusion matrix for borderline-SMOTE + LSTM.

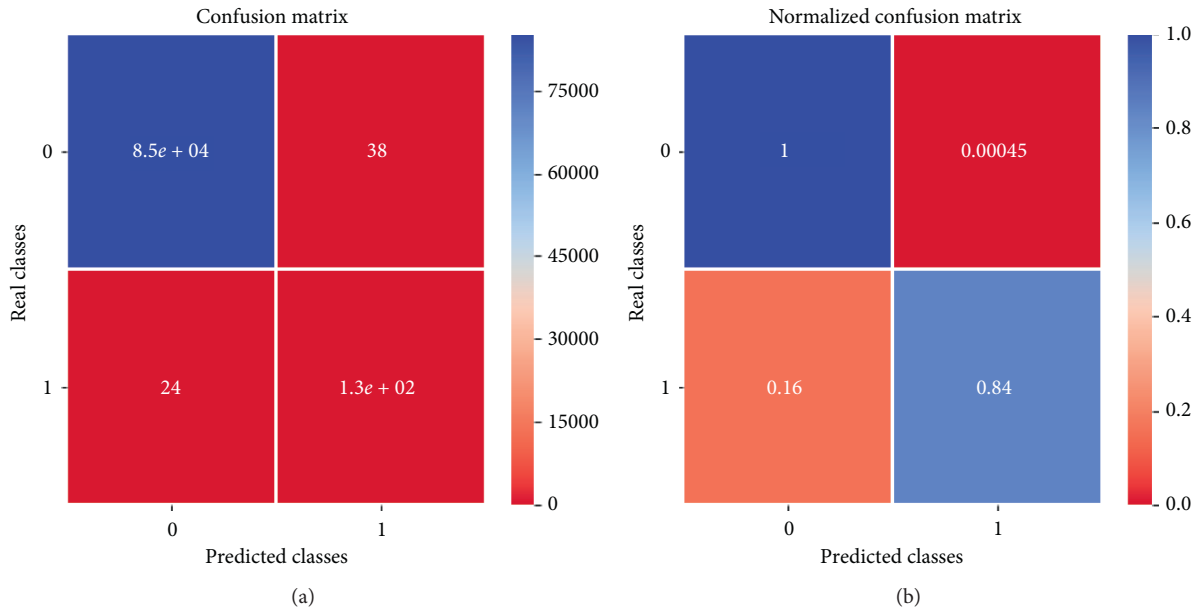

Figure 12: (a) Confusion matrix and (b) normalized confusion matrix for SVM SMOTE + LSTM.

misclassification of majority samples. The misclassification rate of most samples was 0.000082, showing a better classification performance.

To sum up, the kNN-SMOTE-LSTM anomaly detection network model proposed in this paper is an effective method to deal with the anomaly problem of wireless sensor net-

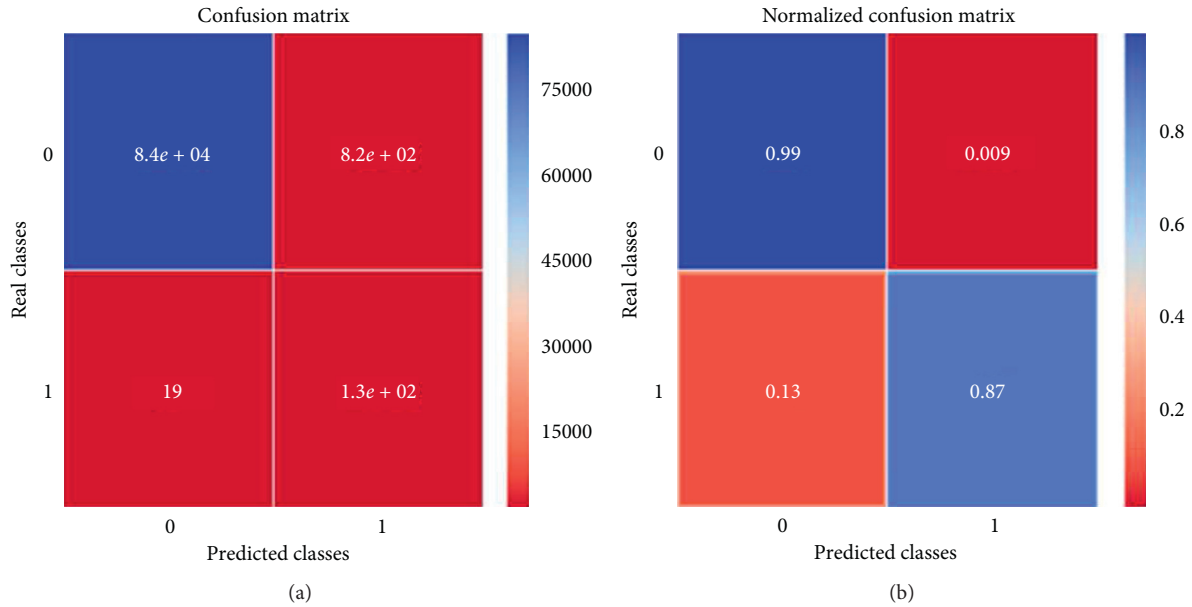

Figure 13: (a) Confusion matrix and (b) normalized confusion matrix for SMOTEENN + LSTM.

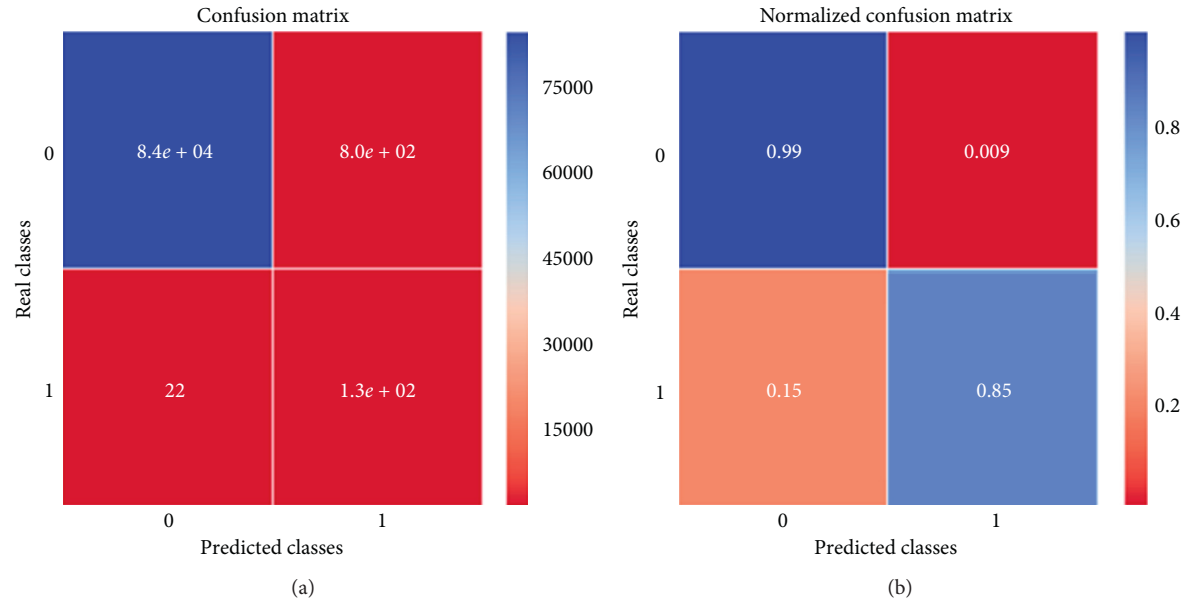

Figure 14: (a) Confusion matrix and (b) normalized confusion matrix for SMOTETomek + LSTM.

work. The KNN-SMOTE-LSTM model can effectively deal with unbalanced data like wireless sensor network anomaly through circulated organic structural fusion. Experiment results show that the classifier based on deep learning model is more applicable in solving complex nonlinear problems like wireless sensor network anomaly.

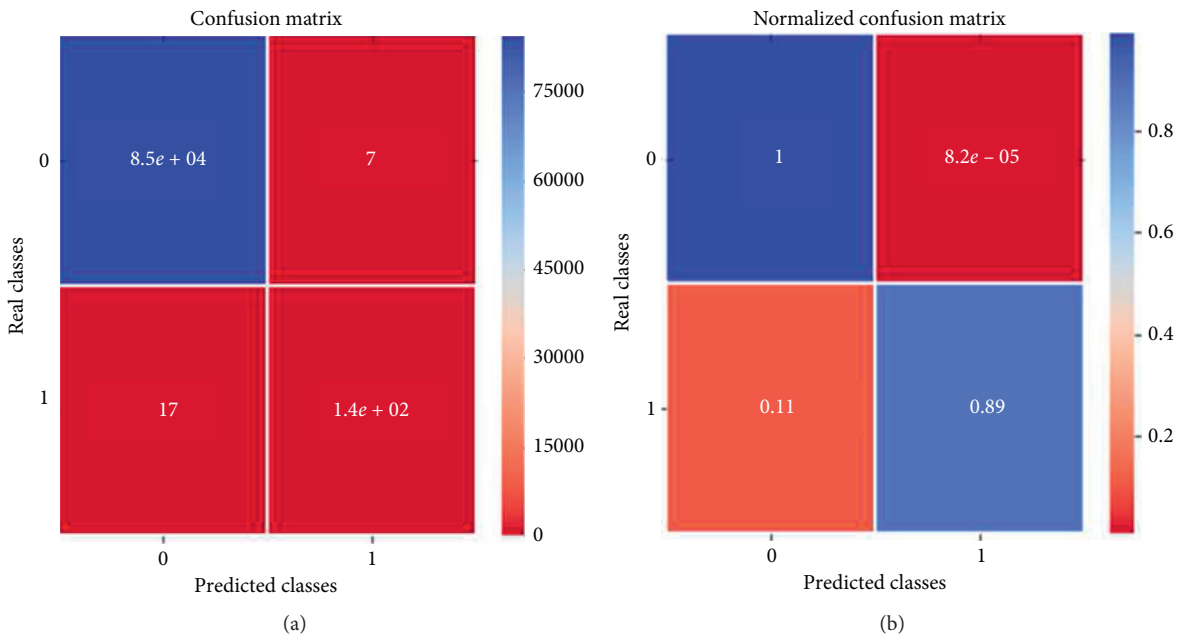

Figure 15: (a) Confusion matrix and (b) normalized confusion matrix for kNN-SMOTE-LSTM.

5. Conclusion and Future Works

In this paper, we mainly focus on the anomaly detection which is a challenging machine learning issue. It presents a WSN anomaly detection model based on kNN-SMOTE-LSTM which is composed of the based classifier, the discriminant classifier, and the data generator through the researches of the unbalanced classification, data mining, and deep learning technology. Experiments demonstrate that, compared with other methods of anomaly detection in wireless sensor network, this model can overcome the defect of misclassification of unbalanced data of the traditional methods. The accuracy of the model reaches 99.97%, and the AUC value reaches 0.9296, which can improve the efficiency of identifying anomaly detection in wireless sensor network and have a significant warning effect.

In this paper, considering that the distribution of the wireless sensor data will change with time and new abnormal situation may appear at any time, it adopts the LSTM-based anomaly detection network model to classify this kind of time-domain sequence data. As the data distribution of wireless sensor data is unbalanced, which means the abnormal data is only a small portion of all daily monitoring data, it applies the SMOTE algorithm to amplify the data to solve the overfitting caused by unbalanced data. As the SMOTE algorithm would produce noise data, influencing the determination of classification boundary, we adopt the discriminant classifier based on kNN algorithm and the basic classifier based on LSTM to screen out the valid samples and remove the noise samples, which can effectively improve the performance and accuracy of classification.

In terms of empirical study, we make an empirical analysis on the wireless sensor network (WSN) anomaly detection model based on the kNN-SMOTE-LSTM. Firstly, the characteristics of parallel distribution are eliminated by data preprocessing. Then, the feasibility, advantages, and disadvantages of each basic classification algorithm are compared by training the basic classifier. The experiment of training the kNN-SMOTE-LSTM anomaly detection network model and comparing with the unbalanced oversampling algorithm of ADASYN, SMOTE, Borderline-SMOTE, svmSMOTE, SMOTEENN, and SMOTETomek combined with the LSTM model demonstrates that the kNN-SMOTE-LSTM anomaly detection model based on basic classifier LSTM, data generator, and discriminant classifier can overcome the defects of misclassification of the traditional method through circulated organic structural fusion. The project aims to improve the existed anomaly detection process, improve the accuracy of anomaly detection, and better interpret the anomaly patterns and prevent anomalies through techniques in the data-driven strategies.

Data Availability

The Oracle data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was funded by the Natural Science Foundation of Zhejiang Province (Grant no. LQ20G010002) and the National Science Foundation of China (71571162). The authors also gratefully acknowledge the helpful comments

and suggestions of Fang Yi, Geyao Li, and Yihao Jiang which have improved the presentation.

References

[1] J. Peng, X. Xiaofeng, Q. Yan et al., "High-dimensional data anomaly detection for WSNs based on deep belief network," *Chinese Journal of Sensors and Actuators*, vol. 32, no. 6, pp. 892–901, 2019.

[2] A. Arranz, M. A. Sanzbobi, and J. Cousstino, "DADICC: intelligent system for anomaly detection in a combined cycle gas turbine plant," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2267–2277, 2008.

[3] H. K. Kim, K. H. Im, and S. C. Park, "DSS for computer security incident response applying CBR and collaborative response," *Expert Systems with Applications*, vol. 37, no. 1, pp. 852–870, 2010.

[4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[5] G. Fortino, R. Giannantonio, R. Gravina, P. Kuryloski, and R. Jafari, "Enabling effective programming and flexible management of efficient body sensor network applications," *IEEE Transactions on Human-Machine Systems*, vol. 43, no. 1, pp. 115–133, 2013.

[6] G. Fortino, A. Guerrieri, G. M. P. O'Hare, and A. Ruzzelli, "A flexible building management framework based on wireless sensor and actuator networks," *Journal of Network and Computer Applications*, vol. 35, no. 6, pp. 1934–1952, 2012.

[7] J. A. Stankovic, "When sensor and actuator networks cover the world," *ETRI Journal*, vol. 30, no. 5, pp. 627–633, 2008.

[8] H. H. Bosman, A. Liotta, G. Iacca, and H. Wortche, "Anomaly detection in sensor systems using lightweight machine learning," in *Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 7–13, IEEE, Manchester, UK, October 2013.

[9] D. Izadi, J. Abawajy, S. Ghanavati, and T. Herawan, "A data fusion method in wireless sensor networks," *Sensors*, vol. 15, no. 2, pp. 2964–2979, 2015.

[10] M. Rassam, A. Zainal, and M. Maarof, "Advancements of data anomaly detection research in wireless sensor networks: a survey and open issues," *Sensors*, vol. 13, no. 8, pp. 10087–10122, 2013.

[11] A. Abduvaliyev, A.-S. K. Pathan, Z. Jianying, R. Roman, and W. Wai-Choong, "On the vital areas of intrusion detection systems in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1223–1237, 2013.

[12] M. Steiger, S. J. Bernard, H. Lücke-Tieke, D. Keim, T. May et al., "Visual analysis of time-series similarities for anomaly detection in sensor networks," *Computer Graphics Forum*, vol. 33, no. 3, pp. 401–410, 2014.

[13] M. Peña, F. Biscarri, J. I. Guerrero, I. Monedero, and C. León, "Rule-based system to detect energy efficiency anomalies in smart Buildings, a data mining approach," *Expert Systems with Applications*, vol. 56, pp. 242–255, 2016.

[14] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer et al., "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, no. 1, pp. 321–357, 2002.

[15] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe-level-smote: safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 475–482, Springer-Verlag, Berlin, Germany, 2009, Advances in Knowledge Discovery and Data Mining.

[16] J. Sun, H. Li, H. Fujita, B. Fu, and W. Ai, "Class-imbalanced dynamic financial distress prediction based on Adaboost-SVM ensemble combined with SMOTE and time weighting," *Information Fusion*, vol. 54, pp. 128–144, 2020.

[17] J. Bi and C. Zhang, "An empirical comparison on state-of-the-art multi-class imbalance learning algorithms and a new diversified ensemble learning scheme," *Knowledge-Based Systems*, vol. 158, pp. 81–93, 2018.

[18] C. Zhang, J. Bi, S. Xu et al., "Multi-Imbalance: an open-source software for multi-class imbalance learning," *Knowledge-Based Systems*, vol. 174, pp. 137–143, 2019.

[19] O. Graa and I. Rekik, "Multi-view learning-based data proliferator for boosting classification using highly imbalanced classes," *Journal of Neuroscience Methods*, vol. 327, Article ID 108344, 2019.

[20] G. Shi, C. Feng, W. Xu, L. Liao, and H. Huang, "Penalized multiple distribution selection method for imbalanced data classification," *Knowledge-Based Systems*, vol. 196, p. 105833, 2020.

[21] G. Fu, Y. Wu, M. Zong et al., "Hellinger distance-based stable sparse feature selection for high-dimensional class-imbalanced data," *BMC Bioinformatics*, vol. 21, no. 1, 2020.

[22] P. Bereziński, B. Jasiul, and M. Szpyrka, "An entropy-based network anomaly detection method," *Entropy*, vol. 17, no. 4, pp. 2367–2408, 2015.

[23] J. J. Flores, F. Calderon, and J. M. A. Garcia, "Network anomaly detection by continuous hidden markov models: an evolutionary programming approach," *Intelligent Data Analysis*, vol. 19, no. 2, pp. 391–412, 2015.

[24] J. Huang, Q. Zhu, L. Yang, and J. Feng, "A non-parameter outlier detection algorithm based on Natural Neighbor," *Knowledge-Based Systems*, vol. 92, pp. 71–77, 2016.

[25] W. N. Robinson and A. Aria, "Sequential fraud detection for prepaid cards using hidden Markov model divergence," *Expert Systems with Applications*, vol. 91, pp. 235–251, 2018.

[26] M. A. Khan, M. R. Karim, and Y. Kim, "A scalable and hybrid intrusion detection system based on the convolutional-LSTM network," *Symmetry*, vol. 11, no. 4, p. 583, 2019.

[27] T. Ergen and S. S. Kozat, "Unsupervised anomaly detection with LSTM neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 8, pp. 3127–3141, 2020.

[28] D. Čeponis and N. Goranin, "Investigation of dual-flow deep learning models LSTM-FCN and GRU-FCN efficiency against single-flow CNN models for the host-based intrusion and malware detection task on univariate times series data," *Applied Science*, vol. 10, p. 2373, 2020.

[29] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," *Neurocomputing*, vol. 262, no. 11, pp. 134–147, 2017.

[30] T. Mikolov, M. Karafiát, L. Burget et al., "Recurrent neural network based language model," in *Proceedings of the 11th Annual Conference of the International Speech Communication Association*, Chiba, Japan, September 2010.

[31] A. M. Schäfer and H.-G. Zimmermann, "Recurrent Neural Networks are universal approximators," *International Journal of Neural Systems*, vol. 17, no. 4, pp. 253–263, 2007.

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] F. A. Gers and E. Schmidhuber, "LSTM recurrent networks learn simple context-free and context-sensitive languages,"

IEEE Transactions on Neural Networks, vol. 12, no. 6, pp. 1333–1340, 2001.

[34] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer Press, Berlin, Germany, 2012.

[35] J. Chung, C. Gulcehre, K. Cho et al., "Gated feedback recurrent neural networks," *Computer Science*, vol. 2015, pp. 2067–2075, 2015.

[36] C. Olah, "Understanding-LSTMs," 2015, http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

[37] Z. LiangH. Shu-Guang et al., "CAPTCHA recognition method based on RNN of LSTM," *Pattern Recognition and Artificial Intelligence*, vol. 24, no. 1, pp. 40–47, 2011.

[38] H. Han, W. Y. Wang, and B. H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," in *International Conference on Intelligent Computing*, pp. 878–887, Springer, Berlin, Germany, 2005, Lecture Notes in Computer Science.

[39] R. Magán-Carrión, D. Urda, I. Díaz-Cano, and B. Dorronsoro, "Towards a reliable comparison and evaluation of network intrusion detection systems based on machine learning approaches," *Applied. Science*, vol. 10, p. 1775, 2020.

[40] C. Xu, "A big-data oriented recommendation method based on multi-objective optimization," *Knowledge-based Systems*, vol. 177, pp. 11–21, 2019.

[41] Z. Qu, H. Li, Y. Wang, J. Zhang, and A. Abu-Siada, "Detection of electricity theft behavior based on improved synthetic minority oversampling technique and random forest classifier," *Energies*, vol. 13, no. 8, p. 2039, 2020.

[42] H. He, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proceedings of the 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 1322–1328, IEEE, Hong Kong, China, June 2008.

[43] E. Zhang, B. Li, P. Li, and Y. Chen, "A deep learning based printing defect classification method with imbalanced samples," *Symmetry*, vol. 11, no. 12, p. 1440, 2019.

[44] C. Ju, F. Bao, C. Xu et al., "A novel method of interestingness measures for association rules mining based on profit," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 868634, 10 pages, 2015.