applied sciences

MDPI

Article

LSTM and Bat-Based RUSBoost Approach for Electricity Theft Detection

Muhammad Adil [1], Nadeem Javaid [2,*], Umar Qasim [3], Ibrar Ullah [4], Muhammad Shafiq [5] and Jin-Ghoo Choi [5,*]

[1] Department of Electrical and Computer Engineering, COMSATS University Islamabad, Islamabad 44000, Pakistan; adilxyz1@gmail.com

[2] Department of Computer Science, COMSATS University Islamabad, Islamabad 44000, Pakistan

[3] Department of Computer Science, University of Engineering & Technology, New Campus, Lahore 54000, Pakistan; umar.qasim@uet.edu.pk

[4] Department of Electrical Engineering, University of Engineering and Technology Peshawar, Bannu 28100, Pakistan; ibrarullah@uetpeshawar.edu.pk

[5] Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Korea; shafiq.pu@gmail.com

* Correspondence: nadeemjavaidqau@gmail.com or nadeemjavaid@comsats.edu.pk (N.J.); jchoi@yu.ac.kr (J.-G.C.)

Received: 30 April 2020; Accepted: 21 June 2020; Published: 25 June 2020

check for updates

Abstract: The electrical losses in power systems are divided into non-technical losses (NTLs) and technical losses (TLs). NTL is more harmful than TL because it includes electricity theft, faulty meters and billing errors. It is one of the major concerns in the power system worldwide and incurs a huge revenue loss for utility companies. Electricity theft detection (ETD) is the mechanism used by industry and academia to detect electricity theft. However, due to imbalanced data, overfitting issues and the handling of high-dimensional data, the ETD cannot be applied efficiently. Therefore, this paper proposes a solution to address the above limitations. A long short-term memory (LSTM) technique is applied to detect abnormal patterns in electricity consumption data along with the bat-based random under-sampling boosting (RUSBoost) technique for parameter optimization. Our proposed system model uses the normalization and interpolation methods to pre-process the electricity data. Afterwards, the pre-processed data are fed into the LSTM module for feature extraction. Finally, the selected features are passed to the RUSBoost module for classification. The simulation results show that the proposed solution resolves the issues of data imbalancing, overfitting and the handling of massive time series data. Additionally, the proposed method outperforms the state-of-the-art techniques; i.e., support vector machine (SVM), convolutional neural network (CNN) and logistic regression (LR). Moreover, the F1-score, precision, recall and receiver operating characteristics (ROC) curve metrics are used for the comparative analysis.

Keywords: non-technical losses; electricity theft; smart meter; random under-sampling; imbalanced data; parameter tuning

1. Introduction

Electricity theft is defined as the consumed amount of energy that is not billed by the consumers. This incurs major revenue losses for electric utility companies [1]. All over the world, electric utility companies lose $96 billion every year due to electricity theft [2]. This phenomenon affects all nations, whether rich or poor. For instance, Pakistan suffers 0.89 billion rupees of loss yearly due to non-technical losses (NTLs) [3] and in India, the electricity loss exceeds 4.8 billion rupees annually [4]. Electricity theft is also a threat to countries with strong economies; i.e., in the U.S., the loss due to electricity theft is approximately $6 billion, and in the UK, it is up to £175 million per annum [5]. In addition, electricity theft causes a voltage imbalance and can affect power system operations by overloading the transformers [6]. Moreover, the rising electricity prices increase the burden on honest customers when the utility asks them also to pay for the theft of energy. It also increases unemployment, the inflation rate and decreases revenue and energy efficiency, which has adverse effects on a country's economical state.

NTL occurs as a result of meter modifications, meter tampering, direct hooking and unregistered connections [7]. The categorization of the NTL is shown in Figure 1. Meter tampering causes the meter either to stop functioning or to stop registering the amount of electricity consumed. In contrast, meter modification is done in the internal settings of a meter to alter its readings. In the direct hooking approach, the consumer taps into a power line from a point ahead of the energy meter, whereas in an unregistered connection, the utility has no record of consumers.

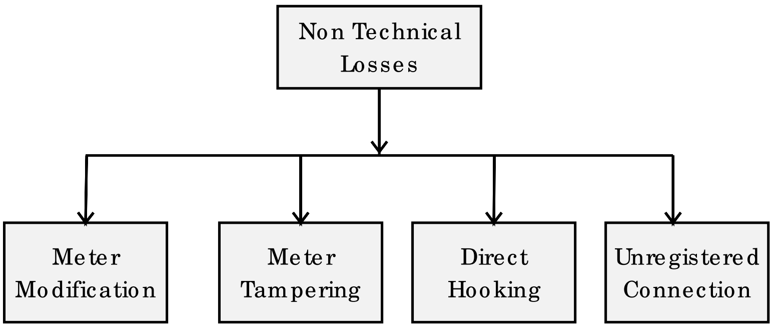

Figure 1. Types of non-technical loss (NTL).

Traditionally, electricity theft has been detected manually by on-field inspections. The inspection teams take meter readings and identify faulty meters for the efficient recovery of the NTL. However, this inspection is time-consuming and requires a separate cost for hiring the inspection teams. In addition, the state-based methods also required hardware installation in the distribution network to detect electricity thefts. With the transitioning of traditional grids into smart grids, smart meters have evolved and their data-driven techniques have contributed to effective energy management [8–10]. The data-driven techniques include machine learning and deep learning algorithms. These algorithms use the procedure of detecting abnormal electricity consumption patterns based on the study of the customers. The machine learning techniques used in the literature are discussed in Section 2. However, most of these approaches have several shortcomings, which are given below.

- Most machine learning techniques require manual feature extraction; as a result, their performance is limited to low-dimensional data and they are not satisfactory for large time series data.
- The problem of class imbalance is a serious concern in electricity theft detection (ETD). In the literature, very little attention has been paid to solving the class imbalance problem.
- The existing machine learning algorithms—i.e., support vector machine (SVM) and logistic regression (LR)—are inefficient in ETD and have a high false positive rate (FPR).
- The state-based solution requires specific hardware devices and has a high cost of installation.
- In most cases, the available dataset has an enormous number of missing values and outliers, which may lead the to the overfitting of the classifier.
- The hyper-parameters of the algorithms are not tuned for optimal classification.

In this paper, we propose a model to address the problems of ETD; i.e., the class imbalanced problem, overfitting, the handling of bigger time series data and the parameter optimization of classifiers. The mapping of the problems addressed and the proposed solution is given in Table 1. The proposed model consists of long short term memory (LSTM) and bat-based random under sampling boosting (RUSBoost) techniques. This work is an extended version of our work in [11].

Conventional models such as the recurrent neural network are hard to train on large electricity consumption data and fail to capture long-term temporal correlations. For this reason, in this paper, we use the LSTM model because it is a sequential model with the long short term memory concept, which shows great significance in the learning of sequential temporal correlations. Moreover, the RUSBoost technique effectively handles the class imbalanced problem and avoids the classifier being biased. In addition, the bat algorithm is used for parameter tuning, finding an optimal learning rate for RUSBoost; this further enhances the performance of the model. This model is efficient at detecting electricity thieves. For validation, the proposed model is compared with state-of-the-art techniques. The simulation results validate the performance of our proposed model. The key contributions of this paper are as follows:

- The smart meter data collected from the State Grid Corporation of China (SGCC) [12] have missing values and outliers. In this paper, we perform data pre-processing using interpolation and normalization methods. These methods help to get the dataset on a common scale and compute the missing values.
- In order to better extract and memorize features from large time series data, we utilize the LSTM block, which efficiently extracts useful information to truly represent electricity theft cases.
- In order to tackle the imbalanced data, RUSBoost is employed to handle the class imbalance problem and performs better than existing data balancing techniques. It performs two operations: RUS first under-samples the data, then Adaboost predicts final classification. This technique improves its performance by learning from previous mistakes, which shows the effectiveness of the model.
- Along with RUSBoost, a metaheuristic method—the bat algorithm—is utilized for the efficient parameter optimization of a classifier.
- Moreover, for comparative analysis, the precision, recall, F1-score and receiver operating characteristics (ROC) curve are used to compute the accuracy of the model.

Table 1. Mapping of problems addressed and proposed solution. RUSBoost: bat-based random under-sampling boosting; LSTM: long short term memory; ROC: receiver operating characteristics.

Limitation Number	Limitation Identified	Solution Number	Proposed Solution
L.1	Imbalanced data	S.1	RUSBoost
L.2	Missing values and outliers	S.2	Interpolation and normalization
L.3	Overfitting	S.3	LSTM and RUSBoost
L.4	High-dimensional data	S.4	LSTM
L.5	Parameter optimization	S.5	Bat optimization
L.6	Reliable Evaluation	S.6	Precision, Recall, F1-Score, ROC

The rest of the work is organized as follows. In Section 2, we present a detailed overview of the literature. In Section 3, the proposed model is described. The experimental results are discussed in Section 4. Finally, the conclusion and future work are given in Section 5 .

2. Literature Review

The existing work related to ETD is classified into three categories: state-based solutions, game theory and machine learning [13]. In particular, the state-based solutions focus on designing specific metering devices and distribution transformers to detect electricity theft [14]. Through the state-based solution, a high detection accuracy can be achieved. However, these methods require additional hardware tools such as meter sensors and distribution transformers, which have a high cost.

The game theory-based solutions [15,16] assume that there is a game between the energy thief and electric utilities. The electricity theft is detected according to the difference between the distribution of electricity consumption that is derived from a game outcome. The game theory-based solutions have a low cost to detect electricity theft. However, it is necessary to define the utility functions for all players in a game, which is time consuming.

The machine learning-based solutions use electricity consumption data to analyze the load profiles of the consumers in order to find benign users. The approaches are further categorized into clustering, semi-supervised and supervised techniques. The clustering techniques can be applied to an unlabelled dataset and rely solely on outlier detection. The authors in [17] presented an unsupervised learning method based on K-nearest neighbors (KNN) to separate anomalous consumption from normal patterns. In semi-supervised learning, both labelled and unlabelled datasets are used for the detection of electricity theft. The authors in [18] proposed a semi-supervised learning technique called the stacked sparse auto-encoder (SSAE) for the detection of NTL in a smart grid. The contributions and drawbacks of the existing techniques are mentioned in Table 2.

Table 2. Contributions and limitations of defense techniques against electricity theft.

Methods	Contributions	Limitations
State-based [14]	State-based solution has achieved a high detection accuracy for electricity theft	High cost of hardware installation
Game theory [15,16]	Game theory-based solutions have a low cost for finding electricity theft	It is necessary to define the utility function for all players in a game, which is time-consuming
Machine learning [17–28]	A data-driven approach is used to effectively detect anomalous consumption behavior	Performance degrades with imbalanced data

The aforementioned methods are related to state-based, game theory and unsupervised machine learning techniques. Our approach proposes a solution based on supervised learning. Therefore, we will study the recent advances made in this area in detail. Some recent studies [19,20] have been based on SVM and LR. The main idea of these methods is to classify honest customers and electricity thieves. The authors in [21] addressed NTL detection in a power distribution system using maximum overlap packet transform (MODWPT) and RUSBoost techniques. The MODWPT method is utilized to extract the relevant features from input data while RUSBoost performs the final classification. In [22], the authors proposed a model based on DT and SVM to detect malicious consumers that intentionally steal electricity; however, no reliable performance metric was used.

In recent times, deep learning approaches have been used in areas such as natural language processing and image recognition. Deep learning techniques are also used to build models to work with the massive data arising from smart meters. They have the ability to learn from huge amounts of data and perform better feature extraction and classification processes. The summary of the existing literature related to supervised machine learning techniques for ETD is presented in Table 3.

Madalina et al. [23] proposed a hybrid neural network composed of LSTM and the multi-layer perceptron (MLP) for ETD. The MLP is used to integrate auxiliary information, while LSTM is used to extract relevant information from the sequential data. The authors in [24] detected electricity theft by extracting the local and global features through a wide and deep CNN. The wide component was used to capture the global features from 1D data, while the deep component was used to capture periodicity from 2D data. Md et al. [25] contributed to the detection of electricity theft by proposing a hybrid model composed of CNN and LSTM. To counter imbalanced datasets, the authors used the synthetic minority over-sampling technique (SMOTE). However, SMOTE generates synthetic data, which causes overfitting and deviates from realistic theft cases. Li et al. [26] developed a hybrid model composed of CNN and random forest (RF) for fraud detection in smart grids; CNN was used to automatically extract features from customers' consumption data, while RF was used to perform the final classification.

The CNN–RF model exhibited very good performance on several performance metrics; i.e., it achieved an area under the curve (AUC) of 97.1% and recall of 96.9%. In [27], an auto encoder was proposed for the detection of anomalies in electricity consumption data using one-dimensional time series data. However, an auto encoder requires hyper-parameters tuning for training.

Ding et al. [28] presented a real-time theft detection approach based on the Gaussian mixture model (GMM) and LSTM. The authors used time series data and enhanced the internal structure of LSTM. The technique was validated in a low-dimensional space and achieved excellent results in terms of the F1-score.

Table 3. Performance of supervised machine learning techniques in the literature. SVM: support vector machine; LR: logistic regression; MODWPT: maximum overlap packet transform; MLP: multi-layer perceptron; AUC: area under the curve; CNN: convolutional neural network; SGCC: State Grid Corporation of China; SMOTE: synthetic minority over-sampling technique; RF: random forest; GMM: Gaussian mixture model.

Techniques	**Dataset**	**Contributions**	**Validation Metrics**	**Limitations**
SVM, DT and LR [19,20]	Smart meter data	To detect the malicious consumers that intentionally steal electricity	Accuracy	No reliable performance metric is used
MODWPT, RUSBoost [21]	Honduras	Achieved better performance when detecting NTL	MCC, F1-score	No parameter tuning
LSTM, MLP [23]	Endesa	Integrate auxiliary information and sequential data effectively to detect electricity theft	ROC, PR AUC	Data imbalance
Wide and deep CNN [24]	SGCC	Capture electricity theft by extracting local and global features from data	AUC, MAP	Data imbalance
SMOTE, CNN and LSTM [25]	SGCC	Improved performance at detecting fraudulent customers	F1-score, MCC	Overfitting
CNN, RF and SMOTE [26]	EISA	Local optima are avoided by using RF in the final layer of CNN	F1-score	High execution time
Auto-encoder [27]	2015 data of Hong Kong	Auto-encoder improves anomaly detection for commercial buildings	Accuracy	Overfitting
LSTM, GMM [28]	Numenta Anomaly Benchmark (NAB)	The internal architecture of LSTM is improved, which enhances the performance compared to the traditional LSTM	F1-score	Not robust

Further research is needed to adequately solve the problems of ETD and overcome the challenges of the poor detection of theft due to imbalanced data and the limited capability of ML algorithms. From the existing literature, we have found that only a few papers have considered the effects of imbalanced data in their system models. The authors in the literature addressed the class imbalance problem by utilizing SMOTE; however, this causes overfitting and replicates the nearest neighbor's samples, which do not reflect real world theft cases. In this paper, we used the RUSBoost method to deal with the imbalanced data. In addition, the traditional machine learning algorithms are used for the classification of ETD; i.e., SVM and LR. However, these methods fail to capture the consumption pattern of large time series data and result in overfitting. Moreover, deep learning techniques have become dominant for capturing real theft cases and have performed better with high-dimensional data. This encouraged us to exploit an LSTM technique to to achieve generalized performance. Additionally, it is necessary to optimize the parameters of classifiers for optimal classification. The authors in the literature have not paid sufficient attention to parameter tuning in their classification; in this regard, we exploit the bat optimization algorithm, which improves the classification of our model.

3. Proposed System Model

The proposed system model for ETD is shown in Figure 2. Our proposed model is mainly composed of three parts: (1) data pre-processing, using interpolation to compute the missing values in the dataset—the data are then normalized and passed to the next model for feature extraction; (2) the LSTM is utilized to extract the relevant features; and (3) the refined features are given to the bat-based RUSBoost algorithm for classification. For comparative analysis, various performance metrics are used—i.e., F1-score, precision, recall and the ROC curve—to validate the effectiveness of our proposed model.

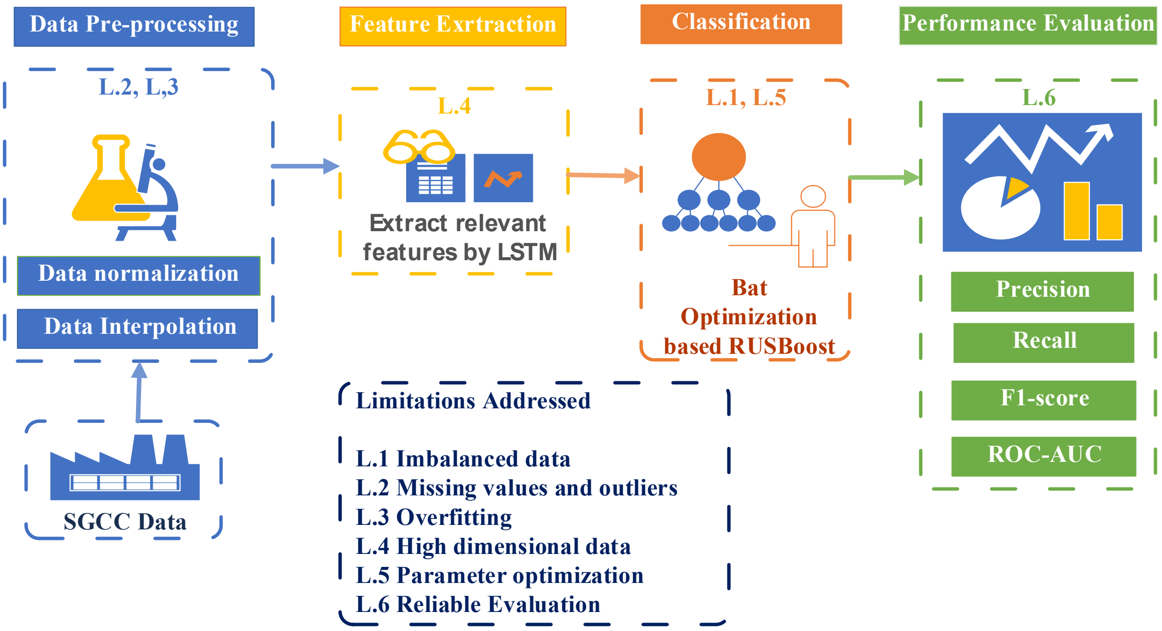

Figure 2. Proposed system model for ETD.

The proposed methodology for ETD is shown in Figure 3 and described in the following subsections.

3.1. Data Pre-Processing

Data pre-processing is relatively important because the real electricity consumption data are inconsistent and often contain missing values. Data pre-processing enhances the performance of the classifier because the performance of the machine learning algorithms depends on the quality of input data. The raw data collected from the SGCC contain many missing and erroneous values, which occur due to the usage of faulty measuring instruments, such as smart meter equipment, or unreliable transmission. The missing values in the dataset mislead the classifier into identifying fraudulent consumers. Additionally, when the data are scattered over a large scale, it makes the analysis difficult and increases the execution time.

We perform two operations in the data pre-processing stage: data interpolation and data normalization. In data interpolation, the missing values are computed and filled using Equation (1), mentioned in [29], as follows:

$$f(x_i) = \begin{cases} \frac{(x_{i+1}+x_{i-1})}{2} & \text{if } x_i \in NaN, x_{i-1}\, and \;\; x_{i+1} \notin NaN \\ 0 & \text{if } x_i \in NaN, x_{i-1}\, or\, x_{i+1} \in NaN \\ x_i & \text{if } x_i \notin NaN, \end{cases} \tag{1}$$

where x_i is the attribute of the electricity consumption data and *NaN* represents the non-numeric value. If both x_{i-1} and x_{i+1} are non-numeric values in the dataset, then they are replaced by zero; otherwise, missing values are replaced by taking the average of the previous and the next values in the dataset.

Afterwards, we use the normalization method to assign a common scale because neural networks are sensitive to diverse data. In the normalization process, the data are scaled in the range from 0 to 1. The values are normalized using Equation (2), mentioned in [30], as follows:

$$A^{'} = \frac{A - Min(A)}{Max(A) - Min(A)} \times ((B - C) + C), \tag{2}$$

where A' is the normalized value while B and C are the maximum and minimum values, respectively. The data normalization facilitates the analysis of data and reduces the model execution time. When A is maximum, then A' = 1. This means that, in the data set, the minimum value is mapped to 0, while the maximum value is mapped to 1.

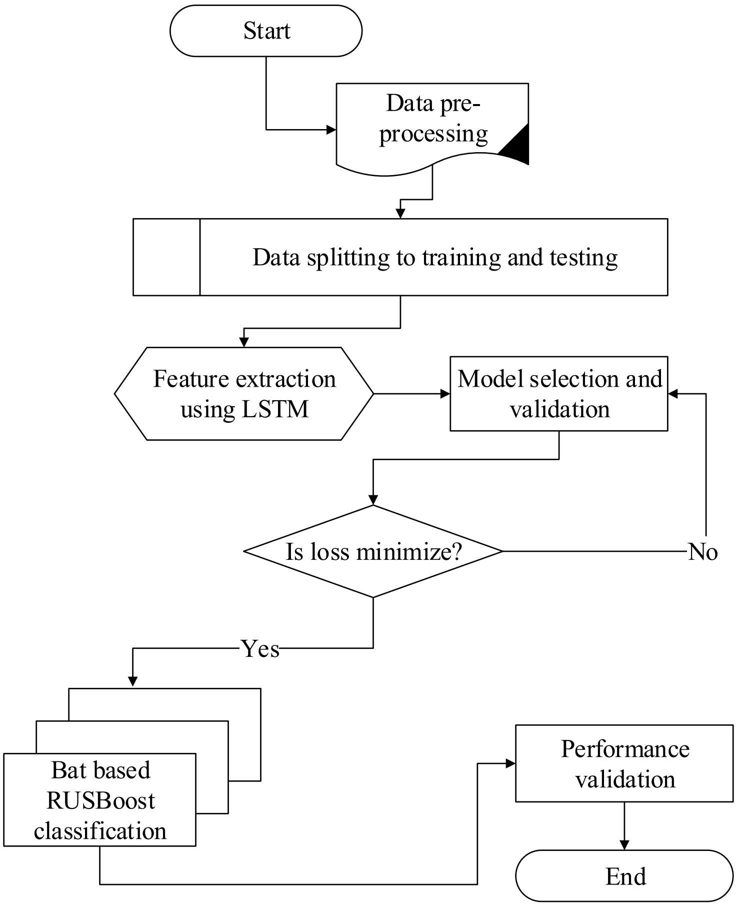
Start
Data pre-processing
Data splitting to training and testing
Feature extraction using LSTM
Model selection and validation
Is loss minimize?
No
Yes
Bat based RUSBoost classification
Performance validation
End

Figure 3. Methodology for ETD.

3.2. Feature Extraction

After the pre-processing stage, the input features are fed into the LSTM module. To get the refined features, the LSTM cell has been used [31]. Since a large amount of data is collected from SGCC, a traditional recurrent neural network (RNN) cannot be adopted. LSTM is a variant of RNN that solves the problems of gradient vanishing and gradient exploding. During the training of RNN, it uses past information and captures a temporal correlation between the previous state and the current input to predict the output. Due to its short memory, RNN fails to regain past information for large time series data.

The LSTM has the ability to capture temporal correlations and classify large time series data. It has been used in many applications as it achieves significant results in speech recognition and image classification problems.

The architecture of the LSTM is shown in Figure 4. It has a special type of memory cells in its architecture that use previous information and memorize the important features from large time series data. The property of the cell state is to keep this information.

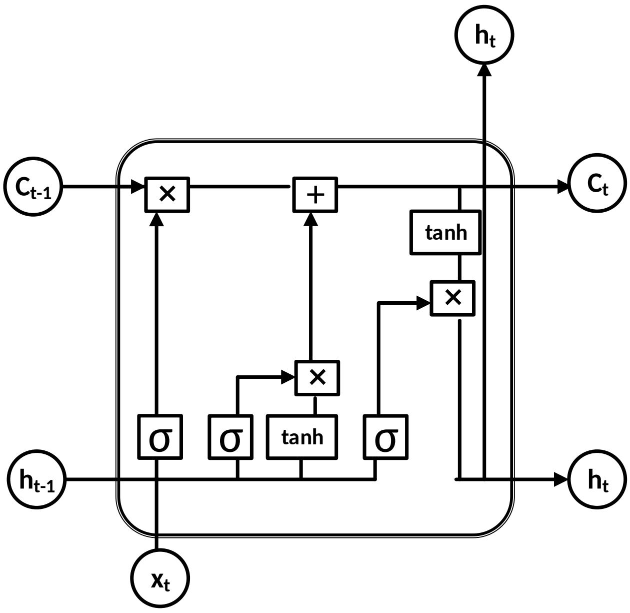

Figure 4. Architecture of LSTM.

The LSTM has three gates: the input gate i_t, the forget gate f_t and the output gate o_t. The forget gate takes the previous hidden state information h_{t-1} and current input x_t through a pointwise multiplication operation and decides either to retain or remove the information from the cell state. This gate uses the sigmoid activation function and predicts an output of either 0 or 1. A value of 1 shows that the relevant information should be kept in the cell state, while 0 represents irrelevant information, which is discarded from the cell state. The forget gate, input gate and output gate are described in Equations (3)–(8), which are mentioned in [25]:

$$f_t = \sigma(W_f[h_{t-1}, x_t]) + b_f. \quad (3)$$

where W_f represents the weight and b_f is the bias of the forget gate f_t. The σ is applied as the activation function on the forget gate.

The input gate decides what information is going to be stored in the cell state C_t. It takes the input x_t and previous hidden state h_{t-1} and applies f_t and *tanh* activation functions through a pointwise multiplication operation as follows:

$$i_t = \sigma(W_i[h_{t-1}, x_t]) + b_i, \quad (4)$$

$$C_t = tanh(W_c[h_{t-1}, x_t]) + b_o. \tag{5}$$

where W_i and W_c represent the weights of the input gate i_t and output gate o_t, respectively. The b_i and b_c are the biases of the network, and C_t is the previous hidden cell state information. To update the information of the current cell state $C_t^{'}$, Equations (4) and (5) are summed through pointwise addition operation, given by Equation (6):

$$C_t^{'} = (f_t \times (C_t)) + (i_t \times (C_t)). \tag{6}$$

Finally, in Equation (7), the output gate is determined. The output gate takes the current input x_t and previous hidden state h_{t-1} with the implication of the activation function σ. The b_o is added as a bias to the output network.

$$o_t = \sigma(W_o[h_{t-1}, x_t]) + b_o. \tag{7}$$

The updated output gate o_t and the information from cell state C_t are used to perform the pointwise multiplication operation to get the next hidden state h_t, given by Equation (8):

$$h_t = \sigma(O_t \times tanh(C_t^{'})). \tag{8}$$

The optimal values are used to obtain better performance for the LSTM. These parameters play an important role in the performance of feature extraction. For better training, we set 50 neurons in each layer except the dense layer, which is a fully connected layer. The dropout is set to 20% in order to avoid the overfitting problem. The hyper-parameter values are shown in Table 4.

Table 4. Hyper-parameter values of LSTM.

Hyper-Parameters	Values
Batch size	50
Drop out	0.2
Optimizer	Adam
Epochs	20
UNITS	50

3.3. Bat Algorithm

Classification accuracy is normally improved through the parameter tunning of the model [32]. We use an optimization technique—the bat algorithm [33]—to choose the best parameter values for RUSBoost. The technique is inspired by the echolocation behavior of bats. During model validation, the hyper-parameters of RUSBoost are tuned to find the optimal values. The hyper-parameters are the learning rate, estimator and sampling strategy. The learning rate is the step size that adjusts the weights of each learner during training abd the estimator is the number of weak learners, which generates the final output.

To find an optimum solution, the bats fly randomly at velocity v_i, frequency f_i and loudness A_i [30]. They utilize an echo system to sense the distance from their prey and find the parameter values for the classifier. The range of frequency is set to [0, f_{max}] for simplicity. The high frequencies cover shorter distances and have shorter wavelengths. Depending on the targets, all the bats adjust their pulse rate in the range [0,1], where 0 means no pulse emission and 1 means the maximum rate of pulse emission.

f_i, v^t and x^t are the updated values of frequency, velocity and position, respectively. The updated f_i, v^t and x^t are shown in Equations (9)–(11), which are mentioned in [34]:

$$f_i = f_{min} + (f_{max} - f_{min})\beta, \tag{9}$$

$$v_i^t = v_i^{t-1} + (x_i - x^*)f_i, \tag{10}$$

$$x_i^t = x_i^{t-1} + v_i^t. \tag{11}$$

where β is a random number ranging between [0,1], while f_{min} is the minimum value of frequency and f_{max} is the maximum value of frequency. In Equation (10), x^* is the best solution in the current population and xi shows the position of a bat at the updated frequency f_i. To find the local position, a solution is selected from the previous best position x_i^{t-1} and by the random flying of a bat at the updated velocity v_i^t. The pseudo-code of the bat algorithm is given in Algorithm 1.

Algorithm 1: Bat algorithm.

```
Initialize bat population by $X_i$ ($i = 1, 2, 3...n$)
Initialize velocity $V_i$, loudness $A_i$ and pulse rate $r_i$
Define the frequency $f_i$, at position $X_i$
Maximum number of iterations is s, and t is the current iteration
while ($t < s$)
    Adjust the $f_i$, $v_i^t$ and $x_i^t$ to find new solutions
    Update the $f_i$, $v_i^t$ and $x_i^t$ as given in Equations (9), (10) and (11)
    if (rand> $r_i$)
        Choose the best solution
        Find the local solution among the selected best solutions
    end if
    Go for new solution
    if (rand<$A_i$ and $f_i < f_*$)
        Select the solution
        Change the loudness $A_i$ and pulse rate $r_i$
    end if
    Rank the bats on the basis of minimum cost function
end if
Generate a final result
```

3.4. Classification of ETD

For the classification of fraudulent and honest consumers, we utilize the RUSBoost technique. As shown in Section 2, the existing models have problems in classifying the ETD. Various random under-sampling (RUS) and random over-sampling (ROS) techniques are used to solve the data imbalance problem. In the RUS technique, data samples from the majority class are discarded to make it equal to the minority class in order to handle the data imbalance issue. However, this technique loses important information from the dataset, which results in a high FPR. Similarly, in the ROS technique, the samples of the minority class are increased by using duplicate information. Therefore, this gives rise to the overfitting problem, and more execution time is required to run the process.

In this paper, we use the RUSBoost method [35], which achieves the benefits of RUS and adaptive boosting techniques. It is efficient for dealing with data imbalance problems. The RUSBoost method first takes random samples from the input data. Then, an ensemble of decision trees is employed, which are weak classifiers. During the training, samples from the majority class are made equal to the minority class. In each iteration, the classification rate of each learner is computed. The instances of theft that are misclassified by the learner have more weight assigned to them. Giving the higher weight to the misclassified instances in the boosting method ameliorates the loss of information in the RUS technique. The final output is obtained through the ensemble of majority learners.

4. Simulation Results and Discussion

The simulation results are explained in this Section. In Section 4.1, we show the dataset information. Section 4.2 shows the simulation environment. Sections 4.3 and 4.4 describe the performance metrics and configuration of benchmark models, respectively. Sections 4.5 and 4.6

present the results of our proposed model and benchmark schemes, respectively. The results are further analyzed in Section 4.7.

4.1. Dataset Information

The electricity theft data are collected from SGCC, which is the largest utility in China. The data are based on electricity consumption data. In this dataset, the data are labeled as either honest or theft. The distribution of classes is imbalanced, and samples are scattered on a large scale. The data also contain missing and erroneous values, which require preprocessing techniques. The released data also provide information about the ground truth, from which 9% of total customers were found to have committed theft, meaning that electricity theft is a severe problem in China. A description of the data is shown in Table 5.

Table 5. Description of data.

Description	Values
Duration of data collection	2014–2016
Number of fraudulent customers	1592
Number of honest customers	8560
Total customers	10,152
Ground truth	9%

4.2. Simulation Environment

We performed the simulations in Python. All the algorithms were trained and built in Keras [36]. The simulations were performed on a platform with an Intel Core i5 with 4 GB RAM. We conducted the simulations on the preprocessed data set, which was pre-processed through the normalization and interpolation methods. The proposed model was trained on the dataset of the SGCC. Firstly, the data were split into groups of 75% and 25% of the data; i.e., 75% of the data was used for training the model and 25% for testing it. For the training of the LSTM model, we passed 20 epochs initially, maintaining a dropout of 0.2 with a batch size of 50 using the Adam optimizer. We used the bat optimization technique to select the optimal hyper-parameters of the RUSBoost model. The configuration of the benchmark models is also given in Section 4.4.

4.3. Performance Metrics

As mentioned in Section 1, the binary classification problems used for detecting electricity theft involve imbalanced data. To evaluate the imbalanced data, the precision, recall and F1-score are quite effective. The metrics used for the evaluation of the proposed model are shown as follows.

4.3.1. F1-Score

The F1-score is widely used for the evaluation of imbalanced datasets. It gives more reliable results than the accuracy score. The F1-score is calculated from the precision and recall. The precision shows the relevancy to the total number of actual results. The precision can be calculated as the number of true positives divided by the sum of true positives and false positives. It is given in Equation (12), which is mentioned in [37]:

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives}. \tag{12}$$

where *True Positives* is the number of dishonest consumers accurately predicted by the classifier, while *False Positives* is the number of honest consumers predicted by the model as thieves.

Recall means how many true positives were found over the predicted result. The recall can be calculated as the number of true positives divided by the sum of true positives and false negatives. This is given by Equation (13), described in [37]:

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives}. \tag{13}$$

where *False Negatives* is the number of dishonest consumers predicted by the model as honest consumers. The F1-score uses both the precision and recall to evaluate the performance of a model. It shows the actual outcome of a model. It can be calculated using Equation (14), which is mentioned in [38,39]:

$$F1 - score = 2 * \frac{Precision \times Recall}{Precision + Recall}. \tag{14}$$

4.3.2. ROC Curve

The ROC curve is effective for evaluating an imbalanced dataset. It is obtained by plotting the FPR against the true positive rate (TPR=. In terms of ETD, the TPR is the count of thefts that are actually found to be suspect, while the FPR is the number of honest consumers counted as theft. The value of the ROC curve ranges from 0 to 1. A classifier that obtains an ROC curve value near to 1 is considered to be a good classifier. The AUC can be calculated using Equation (15), which is mentioned in [40]:

$$AUC = \frac{\sum Rank_{i \in positive\ class} - \frac{M(1+M)}{2}}{M \times N}. \tag{15}$$

where $Rank_i$ represents the rank value of each sample. M shows the number of positive class samples and N shows the number of negative class samples. The AUC is also called the area under the ROC curve.

4.4. Benchmark Models and Their Configurations

In this section, we describe the conventional models which are widely used as classifiers for ETD. The range of hyper-parameter values is applied, and we select optimal values for each base model.

4.4.1. SVM Model

This is a popular classifier, and it was widely used for the ETD in [19,22]. The SVM finds an optimal hyperplane, which maximizes the margin between different classes. The γ and regularization parameters given in Table 6 are important for the selection of an optimal hyperplane to distinguish classes. We choose optimal hyper-parameter values to select the best model of SVM.

Table 6. Hyper-parameter values of SVM.

Hyper-Parameters	Range of Values	Values
γ	2,5,8	2
C	0.001, 0.01, 0.1	0.1

4.4.2. LR Model

This is a supervised learning algorithm which has been widely used for classification in the existing literature [20]. The LR utilizes the same principles as neural networks. The LR for binary classification task is similar to the single hidden layer-based neural network with either a tangent or sigmoid activation function. The tangent activation function value ranges between −1 and 1, while the sigmoid function value ranges between 0 and 1. In the tangent function, a value near to 1 is classified as theft, and a value near to -1 is classified as honest. The hyper-parameters are given in Table 7. During the implementation, we choose optimal values for accurate classification.

Table 7. Hyper-parameter values of LR.

Hyper-Parameters	Range of Values	Values
C	0.001, 0.01, 0.1	0.1
R	l1 norm, l2 norm	l2 norm

4.4.3. Hybrid CNN–LSTM Model

Along with the conventional machine learning algorithms, we have also used a hybrid CNN–LSTM model [25] as a deep learning technique for comparison. The deep learning techniques are used to build models in order to work with the massive data of smart meters. These models have the ability to learn from time series data and extract the relevant features for accurate classification. The CNN is a feed forward neural network, which is mostly used for complex classification problems. In a hybrid CNN-LSTM model, the CNN is used to capture the global features from 1D data, while the LSTM is used to capture periodicity from 2D data. We choose the best values during model validation, which are given in Table 8.

Table 8. Hyper-parameter values of CNN and LSTM.

Hyper-Parameters	LSTM Values	CNN Values
Batch size	50	130
Dropout	0.2	0.01
Optimizer	Adam	Adam
Epochs	20	40

4.5. Performance of LSTM–RUSBoost Model for ETD

In this section, we describe the problems of the conventional models described in Section 2; i.e., the class imbalance problem, overfitting issue, the handling of missing values in the dataset and lack of parameter optimization during classification. Firstly, we present the performance of our proposed model on raw data. Initially, the missing values in the dataset are filled with the interpolation method. Additionally, the data are normalized using the min–max scaling method. The visualization of the data in Figure 5 shows the imbalanced distribution of labels; i.e., the number of thieving users is represented by "1", while "0" shows honest customers.

The consequences of using imbalanced data are that it trains the classifier on a majority dataset, which results in the classifier being biased towards identifying fraudulent electricity consumers. In this regard, the RUSBoost method is effective, as already described in Section 3.4. It achieves the benefits of RUS and adaptive boosting techniques. It is efficient in dealing with data imbalance problems. In order to show the effects of imbalanced data, we also implement SMOTE along with the SVM technique. The authors in [22,23] used SMOTE to solve the data imbalance problem. In the SMOTE technique, the samples of the minority class are increased by taking the information from their nearest neighbor. This gives rise to the overfitting problem, and more execution time is required to run the process.

The training complexity of SVM is highly dependent on the input data. Figure 6 shows that, due to training with imbalanced data, the performance of SVM is poor and it fails to classify the fraud users successfully. The model is trained on a balanced dataset after SMOTE is applied to generate synthetic data. Although the SMOTE–SVM method shows better results in comparison with SVM only, SMOTE adds synthetic data samples to the minority class, and the synthetic data do not represent real theft cases. Comparatively, our proposed method performs better for the imbalanced dataset in terms of all performance metrics.

During model validation, the hyper-parameters of RUSBoost are tuned to find optimal values. The hyper-parameters are the learning rate, estimator and sampling strategy. We optimize the learning rate of RUSBoost. Setting the hyper-parameters manually is a time-consuming task. The bat algorithm is efficient for finding the optimal value of the learning rate. The results in Figure 7 show that parameter tuning enhanced the performance of our proposed model in terms of all performance metrics.

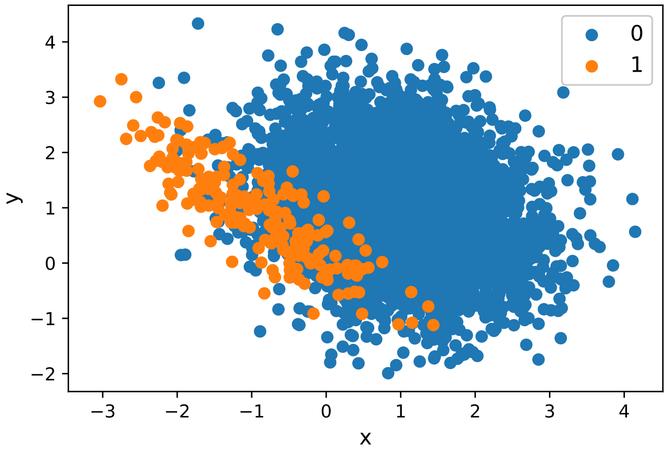

Figure 5. Visualization of unbalanced data.

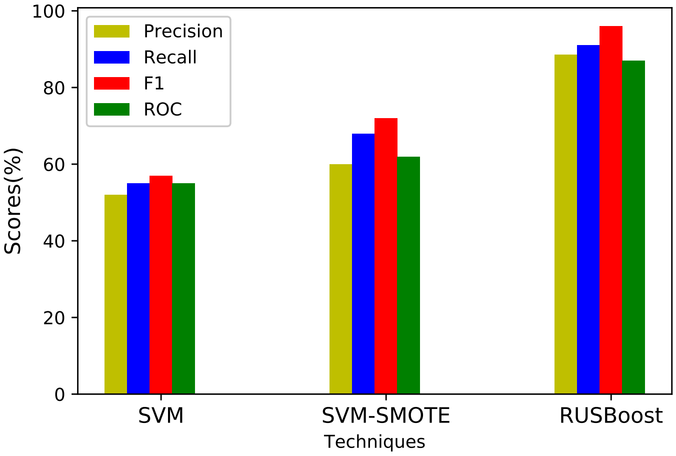

Figure 6. Performance comparison of balanced and unbalanced datasets.

Table 9 shows the confusion matrix of the true negative (TN), true positive (TP), false negative (FN) and false positive (FP) rates of our proposed technique. Our goal in ETD is to maximize the TPR and minimize the FPR. The confusion matrix shows good results for our proposed model. It achieves a high percentage of TPR with a low FPR.

In order to perform a reliable evaluation, we also evaluate our model wit the ROC curve. The ROC curve for our proposed model is shown in Figure 8. It is obtained by plotting the TPR against the FPR. To train and test data, ROC covers a large area under the curve, which shows that the model has accurately predicted the theft cases. Moreover, as mentioned in Section 3, the existing models [19–25] tend to overfit when dealing with imbalanced data. In our proposed model, the results obtained for both training and testing data are good, which means that the model does not overfit out-of-sample data. The mapping of the problems addressed and validation results are given in Table 10.

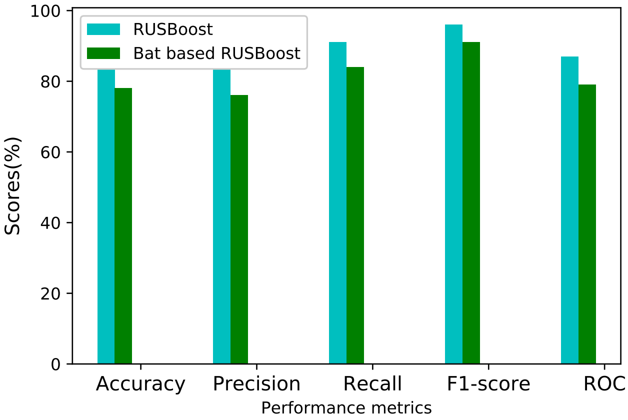

Figure 7. Performance comparison of parameter tuning.

Table 9. Confusion matrix values of the proposed model. TN: true negative; FP: false positive; FN: false negative; TP: true positive.

Confusion Matrix	Predicted No	Predicted Yes
Actual No	TN = 496	FP = 117
Actual Yes	FN = 92	TP = 535

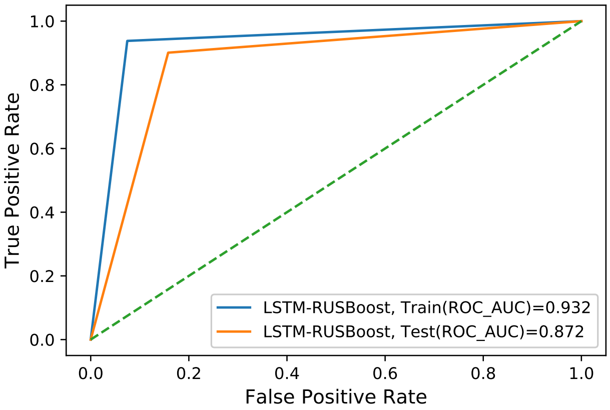

Figure 8. ROC-AUC of proposed model.

Table 10. Mapping of problems addressed and validation results.

Limitation Number	Limitation Identified	Proposed Solution	Validation Results
L.1	Imbalanced data	S.1	RUSBoost classifier effectively handles the imbalanced data as shown in Figure 6
L.2	Missing values and outliers	S.2	No direct validation
L.3	Overfitting	S.3	Our proposed LSTM and bat-based RUSBoost approach obtain generalized performance, as shown in Figure 8
L.4	High-dimensional data	S.4	No direct validation
L.5	Parameter optimization	S.5	The bat algorithm enhances the performance of RUSBoost as shown in Figure 7
L.6	Reliable evaluation	S.6	We obtain a reliable evaluation of our model as indicated in Figures 7 and 8

4.6. Performance Comparison

For comparison, the proposed model is compared with SVM [16], LR [17] and the hybrid CNN–LSTM [22] model, which are the state-of-the-art benchmark models. The details and configurations of these models are given in Section 4.4.

4.6.1. SVM Model Results

The SVM is a popular method used for the classification of ETD. However, it fails to achieve generalized performance. The SVM gives better results for training data. However, for out-of-sample data, the model tends to overfit. The overfitting in SVM for high-dimensional data is evident from Figure 9; i.e., the AUC for the training data is 73.4%, and the AUC for out-of-sample data is only 57.3%. Thus, due to the generalization problem, its performance degrades. The confusion matrix for this model also shows a high FPR, which means that it falsely rejects the theft cases during classification. Table 11 shows the TP, TN, FP, and FN values for SVM. Furthermore, SVM covers less area under the ROC curve than our proposed method. This model has poor performance compared to the other benchmark schemes. Thus, SVM is not suitable for the class imbalance problem considering high-dimensional data.

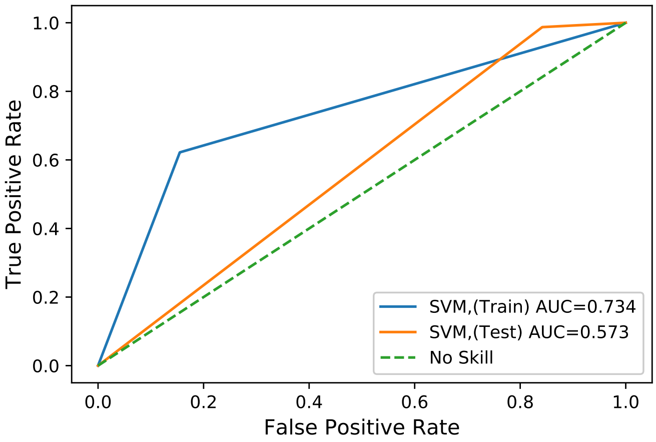

Figure 9. ROC-AUC of the SVM model.

Table 11. Confusion matrix values of the SVM model.

Confusion Matrix	Predicted No	Predicted Yes
Actual No	TN = 97	FP = 516
Actual Yes	FN = 8	TP = 619

4.6.2. LR Model Results

LR uses the principle of neural networks and the logistic sigmoid function to return the value of the variable. It is used for binary classification problems, as already discussed in Section 2. The configuration of the LR model is given in Section 4.4. We implement this model using the SGCC data. Moreover, we have investigated the effects of highly imbalanced data on the performance of the supervised learning LR model. The performance of LR without any class balancing technique is the worst of the models used. We utilized SMOTE for balancing the data and then implemented the LR. The confusion matrix for the LR model is shown in Table 12. This gives the information regarding how accurately the model predicts electricity theft. The algorithm is efficient in predicting the number of honest consumers; however, the FNR is still high, which means that it misses real theft cases and has poor results in detecting the electricity theft. This implies that the LR has poor performance. Figure 10 shows the ROC-AUC of LR model.

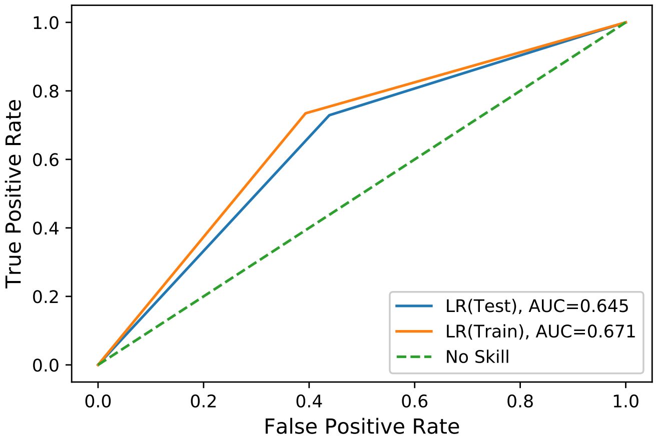

Figure 10. ROC-AUC of LR model.

Table 12. Confusion matrix values of LR model.

Confusion Matrix	Predicted No	Predicted Yes
Actual No	TN 518	FP 95
Actual Yes	FN 237	TP 390

4.6.3. Hybrid CNN-LSTM Model Results

In a hybrid CNN–LSTM model, the CNN is used to capture the global features from one-dimensional data, while the LSTM is used to capture periodicity from two-dimensional data. The hybrid model has the ability to learn from huge amounts of data and perform better for feature extraction and tje classification of electricity theft. To evaluate the CNN–LSTM model, we use the ROC curve, precision, recall and F1-score. The CNN–LSTM model exhibits better results, as shown in Table 13. The CNN–LSTM model performs better as compared to the other two models; i.e., SVM and LR. The small bias in the test dataset in Figure 11 shows that the CNN–LSTM can learn features with a large amount of electricity consumption data. During hyper-parameter tuning, increasing the number of epochs decreases the training and testing loss for the this model. However, this model has a high execution time.

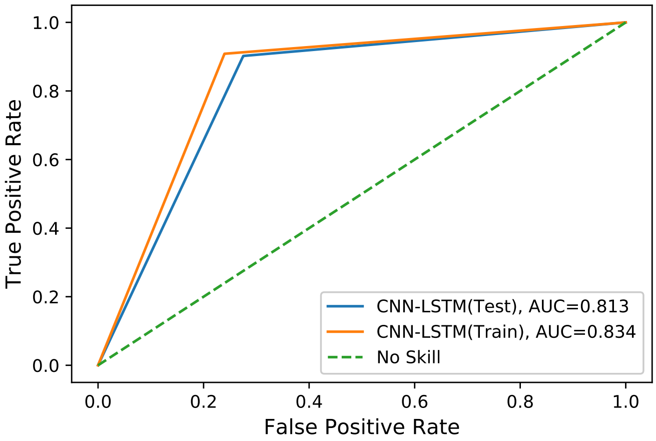

Figure 11. ROC-AUC of CNN–LSTM model.

4.7. Summary of Results

The summary of results is given in Figure 12 and Table 13. In order to validate the performance of the proposed system model, it is compared with the state-of-art benchmark schemes. We use reliable performance metrics such as the precision, recall, F1-score and ROC curve. For comparison, the benchmark models used in this paper are LR, SVM and CNN–LSTM models. The results prove that SVM performs worst compared to the other benchmark schemes. This is due to the fact that SVM does not handle large time series data; for massive data, it causes overfitting. We see from Figure 9 that, for testing data, the result of SVM is shown to be the worst: i.e., it achieves an ROC curve of 57.2%. The CNN–LSTM model shows better performance compared to SVM and LR; i.e., the value of the ROC curve is 81% and recall is 85%. It is considered to be the best model among the benchmark schemes. The CNN–LSTM model performs better because it is a deep learning model and can handle large time series data well.

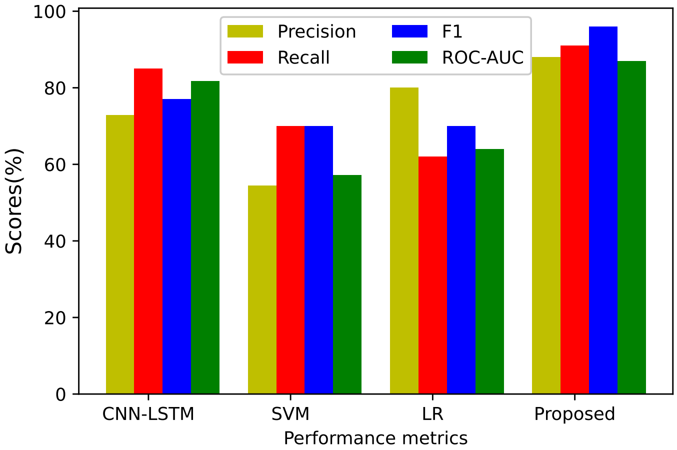

Figure 12. Performance comparison.

The proposed LSTM–RUSBoost model is reliable and beats the given benchmark schemes in terms of all performance metrics. Our proposed model shows superiority to other models for many reasons; firstly, it can effectively handle imbalanced data well by the random under-sampling operation and then by using the adaptive boosting technique for classification. Secondly, the LSTM block efficiently extracts the relevant features during feature refinement. Finally, the optimization by the bat algorithm further improves the performance of our proposed system model.

Table 13. Summary of results.

Models	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**ROC**
CNN–LSTM	0.742	0.725	0.851	0.779	0.817
SVM	0.577	0.545	0.70	0.702	0.572
LR	0.732	0.804	0.622	0.701	0.645
Proposed Model	0.879	0.889	0.9109	0.961	0.879

5. Conclusions and Future Work

In this paper, a model for ETD is proposed and evaluated on a real-time series dataset. In the proposed system, the electricity data are pre-processed to remove null and undefined values using the normalization and interpolation methods. Afterwards, the LSTM is used for feature refinement, which extracts the relevant features from the pre-processed data. Finally, the RUSBoost method is applied to balance the data efficiently; i.e., to classify the data into honest and dishonest customers. To enhance the performance of the RUSBoost method, a bat algorithm is used for parameter optimization. For the evaluation of the proposed model, it is compared with SVM, LR and CNN–LSTM models. The simulation results from the evaluation show the superiority of the proposed model over the existing models in terms of handling imbalanced data, parameter optimization and overfitting. Furthermore, using the performance metrics, the proposed model achieves 96.1% for F1-score, 88.9% for precision, 91.09% for recall and 87.9% for ROC-AUC. However, despite the proposed model outperforming alternative techniques, it is overly sensitive to changes in the input data. In future, electricity datasets for both residential and commercial buildings will be considered.

Author Contributions: M.A. and N.J. proposed and implemented the main idea. U.Q. and I.U. performed the mathematical modeling and wrote the simulation section. M.S. and and J.-G.C. organized and refined the manuscript. All authors together responded to the respectable reviewers' comments. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2016-0-00313) supervised by the IITP (Institute for Information and communications Technology Planning and Evaluation, and in part by the 2020 Yeungnam University Research Grant.

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

References

1. Das, A.; McFarlane, A. Non-linear dynamics of electric power losses, electricity consumption, and GDP in Jamaica. *Energy Econ.* **2019**, *84*, 104530. [CrossRef]
2. Northeast Group LLC. Electricity Theft and Non-Technical Losses: Global Markets, Solutions, and Vendors. 2018. Available online: http://www.northeast-group.com (accessed on 22 February 2020).
3. Hussain, Z.; Memon, S.; Shah, R.; Bhutto, Z.A.; Aljawarneh, M. Methods and Techniques of Electricity Thieving in Pakistan. *J. Power Energy Eng.* **2016**, *4*, 1–10. [CrossRef]
4. Meira, J.A.; Glauner, P.; State, R.; Valtchev, P.; Dolberg, L.; Bettinger, F.; Duarte, D. Distilling provider-independent data for general detection of non-technical losses. In Proceedings of the 2017 IEEE Power and Energy Conference at Illinois (PECI), Champaign, IL, USA, 23–24 February 2017; pp. 1–5.
5. Lewis, F.B. Costly 'Throw-Ups': Electricity Theft and Power Disruptions. *Electr. J.* **2015**, *28*, 118–135. [CrossRef]
6. Bashkari, S.; Sami, A.; Rastegar, M. Outage Cause Detection in Power Distribution Systems based on Data Mining. *IEEE Trans. Ind. Inf.* **2020**. [CrossRef]
7. Messinis, G.M.; Hatziargyriou, N.D. Review of non-technical loss detection methods. *Electr. Power Syst. Res.* **2018**, *158*, 250–266. [CrossRef]
8. Gul, H.; Javaid, N.; Ullah, I.; Qamar, A.M.; Afzal, M.K.; Joshi, G.P. Detection of Non-Technical Losses using SOSTLink and Bidirectional Gated Recurrent Unit to Secure Smart Meters. *Appl. Sci.* **2020**, *10*, 3151. [CrossRef]
9. Abid, S.; Alghamdi, T.A.; Haseeb, A.; Wadud, Z.; Ahmed, A.; Javaid, N. An Economical Energy Management Strategy for Viable Microgrid Modes. *Electronics* **2019**, *8*, 1442. [CrossRef]
10. Mujeeb, S.; Javaid, N. ESAENARX and DE-RELM: Novel Schemes for Big Data Predictive Analytics of Electricity Load and Price. *Sustain. Cities Soc.* **2019**, *51*, 101642. [CrossRef]
11. Adil, M.; Javaid, N.; Ullah, Z.; Maqsood, M.; Ali, S.; Daud, M.A. Electricity Theft Detection using Machine Learning Techniques to Secure Smart Grid. In Proceedings of the 14th International Conference on Complex, Intelligent and Software Intensive System (CISIS'20), Lodz, Poland, 1–3 July 2020.
12. State Grid Corporation of China. Available online: https://www.sgcc.com.cn (accessed on 22 February 2020).
13. Jokar, P.; Arianpoo, N.; Leung, V.C. Electricity theft detection in AMI using customers' consumption patterns. *IEEE Trans. Smart Grid* **2016**, *7*, 216–226. [CrossRef]
14. 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]

15. Wei, L.; Sundararajan, A.; Sarwat, A.I.; Biswas, S.; Ibrahim, E. A distributed intelligent framework for electricity theft detection using benford's law and stackelberg game. In Proceedings of the 2017 Resilience Week (RWS), Wilmington, DE, USA, 18–22 September 2017; pp. 5–11.
16. Jamil, A.; Alghamdi, T.A.; Khan, Z.A.; Javaid, S.; Haseeb, A.; Wadud, Z.; Javaid, N. An Innovative Home Energy Management Model with Coordination among Appliances using Game Theory. *Sustainability* **2019**, *11*, 6287. [CrossRef]
17. Micheli, G.; Soda, E.; Vespucci, M.T.; Gobbi, M.; Bertani, A. Big data analytics: An aid to detection of non-technical losses in power utilities. *Comput. Manag. Sci.* **2019**, *16*, 329–343. [CrossRef]
18. Lu, X.; Zhou, Y.; Wang, Z.; Yi, Y.; Feng, L.; Wang, F. Knowledge Embedded Semi-Supervised Deep Learning for Detecting Non-Technical Losses in the Smart Grid. *Energies* **2019**, *12*, 3452. [CrossRef]
19. Toma, R.N.; Hasan, M.N.; Nahid, A.A.; Li, B. Electricity Theft Detection to Reduce Non-Technical Loss using Support Vector Machine in Smart Grid. In Proceedings of the 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, 3–5 May 2019; pp. 1–6.
20. Buzau, M.M.; Tejedor-Aguilera, J.; Cruz-Romero, P.; Gómez-Expósito, A. Detection of non-technical losses using smart meter data and supervised learning. *IEEE Trans. Smart Grid* **2018**, *10*, 2661–2670. [CrossRef]
21. Avila, N.F.; Figueroa, G.; Chu, C.C. NTL detection in electric distribution systems using the maximal overlap discrete wavelet-packet transform and random under sampling boosting. *IEEE Trans. Power Syst.* **2018**, *33*, 7171–7180. [CrossRef]
22. 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. Inform.* **2016**, *12*, 1005–1016. [CrossRef]
23. Buzau, M.M.; Tejedor-Aguilera, J.; Cruz-Romero, P.; Gómez-Expósito, A. Hybrid deep neural networks for detection of non-technical losses in electricity smart meters. *IEEE Trans. Power Syst.* **2019**, *32*, 1254–1263. [CrossRef]
24. 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. Inform.* **2017**, *14*, 1606–1615. [CrossRef]
25. Hasan, M.; Toma, R.N.; Nahid, A.A.; Islam, M.M.; Kim, J.M. Electricity Theft Detection in Smart Grid Systems: A CNN-LSTM Based Approach. *Energies* **2019**, *12*, 3310. [CrossRef]
26. Li, S.; Han, Y.; Yao, X.; Yingchen, S.; Wang, J.; Zhao, Q. Electricity Theft Detection in Power Grids with Deep Learning and Random Forests. *J. Electr. Comput. Eng.* **2019**, *2019*, 4136874. [CrossRef]
27. Fan, C.; Xiao, F.; Zhao, Y.; Wang, J. Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data. *Appl. Energy* **2018**, *211*, 1123–1135. [CrossRef]
28. Ding, N.; Ma, H.; Gao, H.; Ma, Y.; Tan, G. Real-time anomaly detection based on long short-Term memory and Gaussian Mixture Model. *Comput. Electr. Eng.* **2019**, *79*, 106458. [CrossRef]
29. Drmac, Z.; Gugercin, S. A new selection operator for the discrete empirical interpolation method—Improved a priori error bound and extensions. *SIAM J. Sci. Comput.* **2016**, *38*, 631–648. [CrossRef]
30. Xu, L.; Skoularidou, M.; Cuesta-Infante, A.; Veeramachaneni, K. Modeling tabular data using conditional gan. In *Advances in Neural Information Processing Systems*; NIPS, Vancouver Convention Centre: Vancouver, BC, Canada, 2019; pp. 7333–7343.
31. Kong, W.; Dong, Z.Y.; Jia, Y.; Hill, D.J.; Xu, Y.; Zhang, Y. Short-term residential load forecasting based on LSTM recurrent neural network. *IEEE Trans. Smart Grid* **2017**, *10*, 841–851. [CrossRef]
32. Khalid, R.; Javaid, N. A Survey on Hyperparameters Optimization Algorithms of Forecasting Models in Smart Grid. *Sustain. Cities Soc.* **2020**, *61*, 102275. [CrossRef]
33. Wulandhari, L.A.; Komsiyah, S.; Wicaksono, W. Bat algorithm implementation on economic dispatch optimization problem. *Procedia Comput. Sci.* **2018**, *135*, 275–282. [CrossRef]
34. Yang, X.S. A new metaheuristic bat-inspired algorithm. In *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 65–74.
35. Seiffert, C.; Khoshgoftaar, T.M.; Van Hulse, J.; Napolitano, A. RUSBoost: A hybrid approach to alleviating class imbalance. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2009**, *40*, 185–197. [CrossRef]
36. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv* **2016**, arXiv:1603.04467.
37. Hand, D.; Christen, P. A note on using the F-measure for evaluating record linkage algorithms. *Stat. Comput.* **2018**, *28*, 539–547. [CrossRef]

38. Gu, Y.; Cheng, L.; Chang, Z. Classification of Imbalanced Data Based on MTS-CBPSO Method: A Case Study of Financial Distress Prediction. *J. Inf. Process. Syst.* **2019**, *15*, 682–693.
39. Douzas, G.; Bacao, F.; Fonseca, J.; Khudinyan, M. Imbalanced Learning in Land Cover Classification: Improving Minority Classes' Prediction Accuracy Using the Geometric SMOTE Algorithm. *Remote Sens.* **2019**, *11*, 3040. [CrossRef]
40. Smith, A.M.; Lampinen, J.M.; Wells, G.L.; Smalarz, L.; Mackovichova, S. Deviation from Perfect Performance measures the diagnostic utility of eyewitness lineups but partial Area Under the ROC Curve does not. *J. Appl. Res. Mem. Cogn.* **2019**, *8*, 50–59. [CrossRef]