



Article

Detecting Nontechnical Losses in Smart Meters Using a MLP-GRU Deep Model and Augmenting Data via Theft Attacks

Benish Kabir ¹, Umar Qasim ², Nadeem Javaid ^{1,*}, Abdulaziz Aldegeishem ³, Nabil Alrajeh ⁴
and Emad A. Mohammed ⁵

¹ Department of Computer Science, COMSATS University Islamabad, Islamabad 44000, Pakistan

² Department of Computer Science, University of Engineering and Technology at Lahore (New Campus), Lahore 54000, Pakistan

³ Department of Urban Planning, College of Architecture and Planning, King Saud University, Riyadh 11574, Saudi Arabia

⁴ Department of Biomedical Technology, College of Applied Medical Sciences, King Saud University, Riyadh 11633, Saudi Arabia

⁵ Department of Engineering, Faculty of Science, Thompson Rivers University, 805 TRU Way, Kamloops, BC V2C 0C8, Canada

* Correspondence: nadeemjavaidqau@gmail.com

Abstract: The current study uses a data-driven method for Nontechnical Loss (NTL) detection using smart meter data. Data augmentation is performed using six distinct theft attacks on benign users' samples to balance the data from honest and theft samples. The theft attacks help to generate synthetic patterns that mimic real-world electricity theft patterns. Moreover, we propose a hybrid model including the Multi-Layer Perceptron and Gated Recurrent Unit (MLP-GRU) networks for detecting electricity theft. In the model, the MLP network examines the auxiliary data to analyze nonmalicious factors in daily consumption data, whereas the GRU network uses smart meter data acquired from the Pakistan Residential Electricity Consumption (PRECON) dataset as the input. Additionally, a random search algorithm is used for tuning the hyperparameters of the proposed deep learning model. In the simulations, the proposed model is compared with the MLP-Long Term Short Memory (LSTM) scheme and other traditional schemes. The results show that the proposed model has scores of 0.93 and 0.96 for the area under the precision–recall curve and the area under the receiver operating characteristic curve, respectively. The precision–recall curve and the area under the receiver operating characteristic curve scores for the MLP-LSTM are 0.93 and 0.89, respectively.

Keywords: deep learning; GRU; healthcare; MLP; non-technical losses; PRECON; smart cities; smart grids; smart meters



Citation: Kabir, B.; Qasim, U.; Javaid, N.; Aldegeishem, A.; Alrajeh, N.; Mohammed, E.A. Detecting Nontechnical Losses in Smart Meters Using a MLP-GRU Deep Model and Augmenting Data via Theft Attacks. *Sustainability* **2022**, *14*, 15001. <https://doi.org/10.3390/su142215001>

Academic Editor: Nien-Che Yang

Received: 26 October 2022

Accepted: 10 November 2022

Published: 13 November 2022

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



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Background

One of the major achievements of smart grids was the development of the Advanced Metering Infrastructure (AMI) system [1]. This system reduces the danger associated with electricity theft by using its fine-grained computations and tracing ability [2]. However, an increase in the system's usage increases energy theft and consequently leads to a loss of electricity [3]. The loss of electricity is among the problems that reduce the performance of the power grids. There are two types of electricity losses. The first are known as Technical Losses (TLs) and the second are known as Non-Technical Losses (NTLs) [4]. The electric heating of resistive components in transformers, transmission lines, and other types of equipment causes TL, while electricity thefts, billing mistakes, and meter faults are the most common cause of NTL [3]. Electricity companies are particularly interested in reducing NTLs, since it accounts for a significant portion of the overall energy losses. Energy theft is the major type of NTL that involves bypassing meters, modifying the meter's readings, etc. The Electricity Consumption (EC) behavior of users may vary from customer

to customer. Nonetheless, identifying NTL patterns among all of the usual patterns of EC is a crucial task. In order to capture different types of NTL behaviors, handcrafted feature engineering approaches have been used. However, these approaches are costly as well as time-consuming due to their reliance on expert knowledge [3].

On the one hand, energy theft has resulted in losses of more than 20% of India's total energy supply and 16% of China's accumulative energy supply [5]. On the other hand, financial losses due to energy theft are approximately 100 million and 6 billion dollars per year for Canada and USA, respectively [6], while Pakistan faces an annual loss of approximately 0.89 billion rupees as a result of NTLs [7]. Theft of energy has long been a severe problem in conventional power networks worldwide. Different users show different patterns of Electricity Consumption (EC). Nonetheless, distinguishing NTL patterns from regular EC patterns is challenging. To detect and address these NTLs, many approaches are employed [8,9]. These approaches are classified into three fundamental groups: hybrid-oriented, network-oriented, and data-driven-oriented detection systems. The data-driven methods have attracted the attention of academics and research scholars for performing Electricity Theft Detection (ETD) over the last few years.

The data-driven method is composed of machine learning-based classifiers that are used to detect NTLs [7]. These solutions are also used in various fields like healthcare, education, and transport. In [10], deep learning models were trained as binary classifiers to detect energy thefts. The authors investigated several deep learning models, such as the Convolutional Neural Network (CNN), Multi-Layer Perceptron (MLP), Long-Short Term Memory (LSTM), and Gated Recurrent Unit (GRU) networks. However, due to inefficient tuning of hyperparameters, these models exhibit poor generalization. To tackle the generalization issue, previous studies used the Grid Search Algorithm (GSA) to tune the hyperparameters of the models. However, the GSA requires high computational resources to find the optimal combination of parameters.

According to [11,12], ensemble models fail to identify diverse theft patterns of EC due to a significant imbalance in data, resulting in a high False Positive Rate (FPR). Therefore, we propose the use of a hybrid of neural networks referred as MLP-GRU to detect energy theft. Actual smart meter data and auxiliary information from the consumers are used for the data analysis.

The authors of [12] conducted a detailed analysis of ensemble models based upon boosting and bagging methods. They observed that the Random Forest (RF) model obtained the highest DR and the lowest FPR. Moreover, the authors implemented two data balancing techniques, i.e., the Synthetic Minority Oversampling Technique (SMOTE) and near-miss, to compare both oversampling and undersampling algorithms. However, there may be an increase in the chances of overlapping classes when using SMOTE, as it can increase the existence of noise. The problem of anomaly detection was addressed in [13]. In the proposed work, the authors used a deep learning approach this is capable of distinguishing between regular and anomalous consumption patterns. They also handled the drift concept by discriminating between nonmalicious and real anomalies. However, there is a substantial delay between the occurrence of an anomaly and its detection in the proposed approach.

Existing Machine Learning (ML) algorithms require an equal number of instances for each class during model training. For minority classes, these models have a poor predictive performance. For the detection of electricity thefts, there is a lack of theft data in the real world. Therefore, we synthetically generated the theft data using data balancing techniques [14]. Many studies have used different balancing techniques; however, such techniques have a high computational time and executional complexity. In [14], the authors proposed a hybrid technique, K-SMOTE, for data balancing. In the model, a k-means clustering algorithm is used to determine k clusters for abnormal samples. Afterwards, SMOTE is applied on the clusters of theft samples for interpolation to balance the complete data. Based on the balanced data, Random Forest (RF) classification is performed to detect electricity theft behavior. However, to determine optimal values of k and perform tuning of other hyperparameters for data balancing, an optimization algorithm is required.

With the emergence of smart meters, diverse types of energy theft cases have been introduced, and these are difficult to detect using the existing techniques. The authors of [15] presented a statistical and ML-based system designed to identify and alert customers about energy theft. In previous studies, several data-driven techniques for the NTL identification issue have been used. The majority of these studies have concentrated on boosting approaches while ignoring bagging methods, such as Extra Trees (ET) and RF. Furthermore, ML models, such as the Support Vector Machine (SVM) and neural networks, have high FPR values and low detection rates. Neural networks were used in [16] for the prediction of coalbed methane well production.

In [17], the authors employed an Extreme Gradient Boosting (XGBoost) technique to classify the malicious users. However, because of the imbalanced dataset, this technique has a high FPR and requires more onsite inspections. The authors of [18] introduced a boosting method called the Gradient Boosting Theft Detector (GBTD), which is based on three existing boosting models: XGBoost, light gradient boosting, and categorical boosting.

The data-driven methods can be broken down into nonsupervised and supervised learning. The nonsupervised learning techniques have acquired significant attention for their use in identifying energy theft nowadays. However, on big datasets, these techniques lack generalization and can also lead to high FPR values due to the fluctuations in load patterns. The authors of [19] exploited an unsupervised learning model called the Stack Sparse Denoising Auto-Encoder (SSDAE) detector, which extracts abstract features from large datasets. However, auto-encoders tune many hyperparameters, thereby consuming more processing time. Moreover, the SSDAE detector must be rectified regularly with incoming training samples. In [20], the authors introduced a novel solution to data augmentation and relevant feature extraction from high dimensional data using a Conditional Variational Auto-Encoder (CVAE) in conjunction with a CNN classifier.

Various experiments on energy theft identification in AMI have been carried out using ML techniques. The authors of [21] presented an unsupervised learning based anomalous pattern recognition technique to identify energy theft in data streams provided by smart meters. The technique only uses regular consumer usage data for model training. However, the classifier may recognize high energy usage patterns over weekdays and holidays. Furthermore, in [22], the authors proposed a Consumption Pattern-Based Energy Theft Detection (CPBETD) approach to leverage the predictability of consumers' benign and fraudulent class samples. However, the SVM misclassification rate limited the DR, resulting in a high FPR.

Most researchers have focused on EC nonmalicious patterns [23]. However, previous studies have shown poor detection rates and accuracy regarding NTL detection. In [23], the authors developed a hybrid K-means-DNN approach, which is a combination of the K-Nearest Neighbor (KNN) and Deep Neural Network (DNN). The approach detects electricity theft in power grids. However, its detection performance is low. The authors of [24] suggested a hybrid method that enhances the internal structure of the standard LSTM model combined with the Gaussian Mixture Model (GMM). However, the proposed method is applicable only for low dimensional space data and is not very robust. In [25], the authors proposed a hybrid technique based on the SVM and Decision Tree (DT) for detecting illegal consumers. However, no effective performance measures were used for the combined technique's evaluation.

Contribution List

The key contributions of this paper are as follows:

- A hybrid model, referred as MLP-GRU, that identifies NTLs using both metering data and auxiliary data is proposed.
- A data augmentation technique is used due to the scarcity of theft samples. This study uses six theft scenarios to create synthetic instances of EC by modifying the honest samples.

- Meanwhile, a Synthetic Minority Oversampling Technique (SMOTE) is employed to maintain a balance between synthetic and benign samples.
- An optimization algorithm, known as the Random Search Algorithm (RSA), is used to effectively tune the MLP-GRU model's hyperparameters.

The rest of the manuscript is structured as follows. A detailed discussion of the proposed model is provided in Section 2. Afterwards, performance evaluation metrics are described in Section 3. Section 4 discusses the simulation results, while the conclusion of the paper is given in Section 5.

2. Proposed System Model

The proposed work is an extended version of [26]. The model proposed for detecting electricity theft includes two stages: training and testing. These two stages are generally comprised of five major steps. Figure 1 depicts the complete methodology outline of this study.

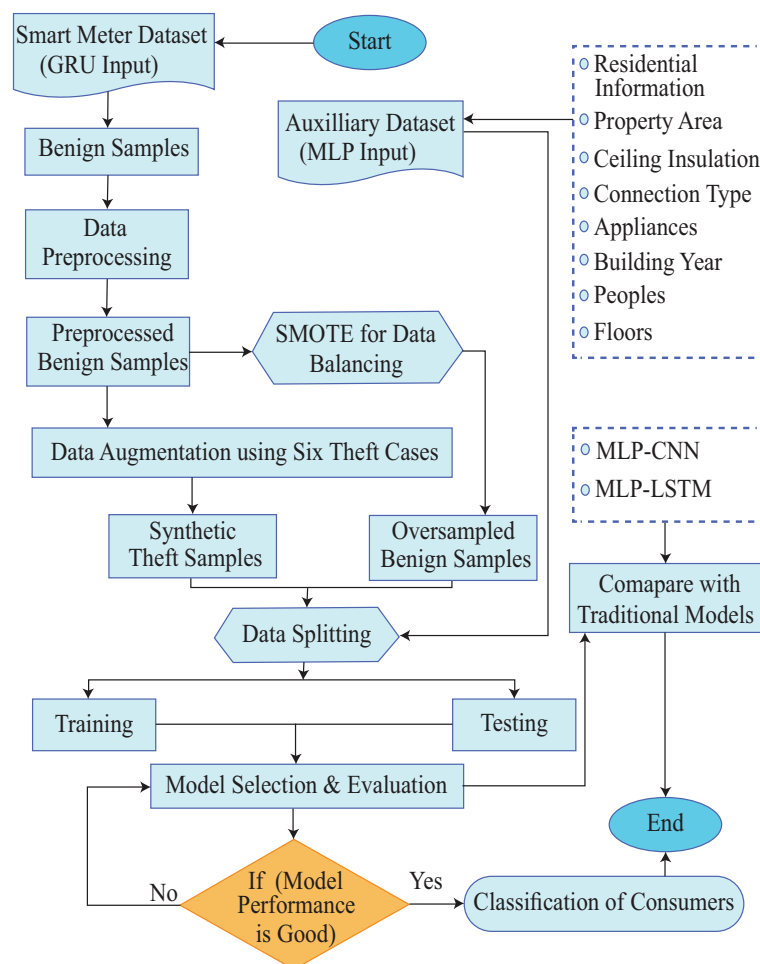


Figure 1. Methodology outline.

- (1) The data preprocessing take place before the training step in the first stage. The data interpolation method is employed to fill in the dataset's missing values. Following that, a standard-scalar technique is used to normalize the data, which is a min-max procedure.
- (2) Data augmentation is performed after the data have been standardized and cleaned. Different theft patterns are created by modifying the honest users' samples using six theft scenarios [18].
- (3) Since the proportion of the theft class exceeds the benign class, SMOTE is applied on the benign class to balance the dataset.

- (4) Afterwards, the preprocessed data are used to train the model. The datasets from the smart meters and relative auxiliary information are sent to the GRU and MLP networks, respectively. The RSA is used to effectively tune the parameters of the classifiers.
- (5) In the last step, efficient performance metrics, such as the accuracy, F1-score, Area Under the Receiver Operating Characteristics Curve (AUC-ROC) and Area Under the Precision–Recall Curve (PR-AUC) are used for evaluating the proposed model’s performance.

During the second stage, we validated the model’s performance by evaluating the unseen data to identify whether the new data belonged to the benign class or the theft class. These steps are shown in Figure 2 and are discussed in the following subsections.

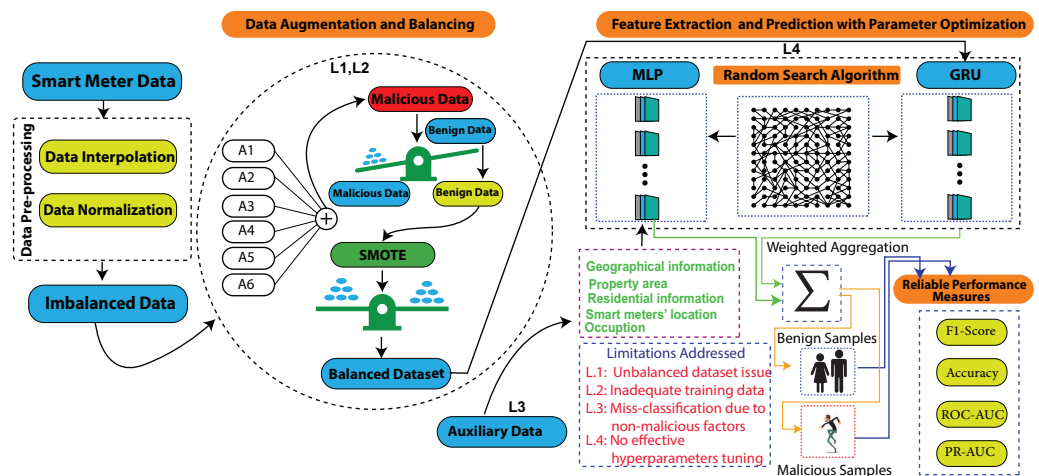


Figure 2. MLP-GRU model architecture.

2.1. Data Preprocessing

The EC data typically contain missing or incorrect numbers due to erroneous data transmission, short circuits in transmission equipment, smart meter failure, and storage problems. The classifier wrongly classifies fraudulent consumers due to missing data in the dataset. We applied an interpolation approach, the simple imputer, to fill in the missing values in the dataset [10]. It was used to impute missing data using the mean, median and so on.

Furthermore, data interpretation becomes complex when the data are spread across a vast scale as the execution time grows. Thus, we normalized the data through a standard-scalar technique, which was used to scale inconsistent data within 0 and 1 to improve the prediction models.

2.2. Data Balancing and Data Augmentation

In the real world, there are fewer nonhonest users’ consumption samples as compared with the amount of benign users’ samples. ML or deep learning models are biased towards majority class samples during training when the dataset is imbalanced. Moreover, they fail to recognize minority class instances that lead to performance degradation.

To address this issue, a variety of resampling techniques have been proposed in the literature [3,17,20]. Undersampling techniques result in the loss of critical data. In contrast, oversampling approaches replicate samples that are likely to be overfitted. The authors of [27] used the One-Dimensional-Wasserstein Generative Adversarial Network (WGAN), which takes a significant amount of time to generate synthetic patterns. Given the significant disparity between massive datasets of energy used and the shortcomings of previous methods, we created synthetic theft instances by altering benign samples in our proposed study. As shown in Figure 3, the Pakistan Residential Electricity Consumption (PRECON) dataset only contains normal users’ samples. Electricity theft samples are also needed

for training the deep learning classifiers to detect electricity theft. Thus, we performed data augmentations through synthetic theft attacks to get nonhonest users' patterns. The samples from fraudulent users were created by modifying the samples of normal users using the six theft attacks. The distribution of augmented data samples is depicted in Figure 4. The six existing theft cases were used to produce distinct malicious patterns using normal ones to train the deep learning classifiers with various theft patterns [10]. The generation of distinct theft patterns to provide diversity in the dataset is an essential feature.

Subsequently, SMOTE was employed to balance the minority class (benign) and majority class (theft) samples. Figure 5 shows the distribution of balanced data using SMOTE. When we generate malicious samples in the dataset, the proportion in the theft class exceeds the benign class. Therefore, we applied SMOTE to the benign class to balance the dataset. In this case, training ML or deep learning models on imbalanced datasets biases the model towards the majority class and adversely affects the model's performance. Thus, oversampling was performed on the data points of the benign class using SMOTE to balance the generated theft instances for each day.

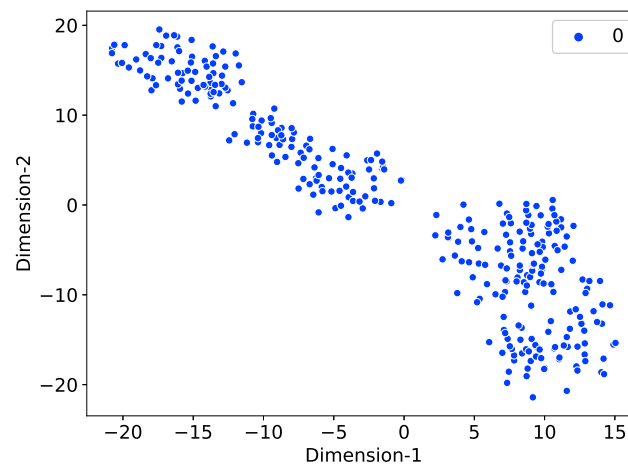


Figure 3. Imbalanced data distribution (Benign class).

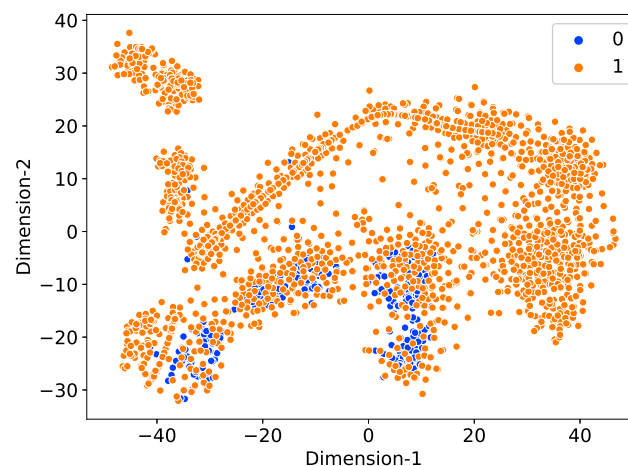


Figure 4. Augmented data using attacks.

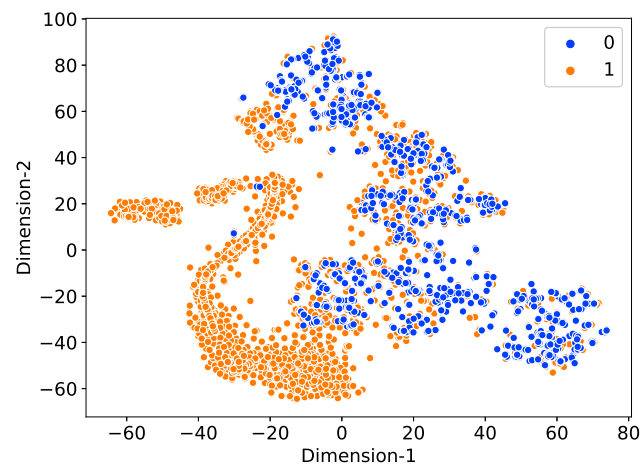


Figure 5. Balanced data distribution.

2.2.1. Six Theft Cases

Existing theft scenarios were used for generating theft data from different attacks by modifying the smart meters' data [18]. In the proposed model, we represent the real daily energy consumption of a home H as H_t , where $H_t = [H_1, H_2, H_3, \dots, H_{48}]$ and $T = 48$ (total actual energy usage per day). We used these theft scenarios to modify the actual energy usage behavior, where t belongs to $[1, 48]$

- (A1). $H_t = H_t * a$, where $a = \text{rand}(0.1, 0.9)$,
- (A2). $H_t = H_t * b_t$, where $b_t = \text{rand}(0.1, 1.0)$,
- (A3). $H_t = H_t * c_t$, where $c_t = \text{rand}[0, 1]$,
- (A4). $H_t = \text{mean}(H) * d_t$, where $d_t = \text{rand}(0.1, 1.0)$,
- (A5). $H_t = \text{mean}(H)$,
- (A6). $H_t = H_{T-t}$.

The first theft attack produces fraudulent patterns by multiplying the consumption of honest users with values randomly produced within the range of 0.1 to 0.9. In theft case 2, each consumer's meter reading is multiplied by a distinct random integer, ranging from (and including) 0.1 to 1.0. The generated values show a discontinuity in tracing the theft data and the manipulated values.

Theft case 3 is an on-off attack in which a consumer either submits the actual readings or a zero value is submitted as its EC. This means that the normal users' samples are multiplied by 1 during a random period t ; otherwise, they are multiplied by zero. Furthermore, for theft attack 4, the average energy consumed for all users is multiplied with a randomly generated value in the range of 0.1 to 1.0 exclusively. As a result, the malicious users under-report the actual energy they consumed. For theft attack 5, the average energy consumed by all users is reported and is the same throughout the day. Theft case 6 changes the sequence of the real EC, for example, by shifting the order of consumption data from peak to off-peak hours [14].

The daily energy usage patterns and six distinct forms of theft cases are shown in Figures 6 and 7.

2.2.2. Hybrid MLP-GRU Network

The hybrid neural network, MLP-GRU, introduced in this work aims to integrate the metering data and auxiliary information. Table 1 shows the auxiliary dataset features with their descriptions. Our proposed method was influenced by the work undertaken in [4] to identify electricity theft, where the authors proposed a hybrid deep neural network, MLP-LSTM. In the proposed model, the GRU network receives the preprocessed EC data from the smart meters. It generalizes the embedding for a shorter processing time by employing few cells. Meanwhile, the auxiliary dataset is provided as an input for the MLP

using 20 neurons. This design is highly efficient, since it allows simultaneous training on both forms of input data. Afterwards, the batch normalization layer is used to normalize the data until it is submitted to the final layer. In the model, the sigmoid activation function in the last layer only has one neuron. The subsections below provide a thorough description of each network.

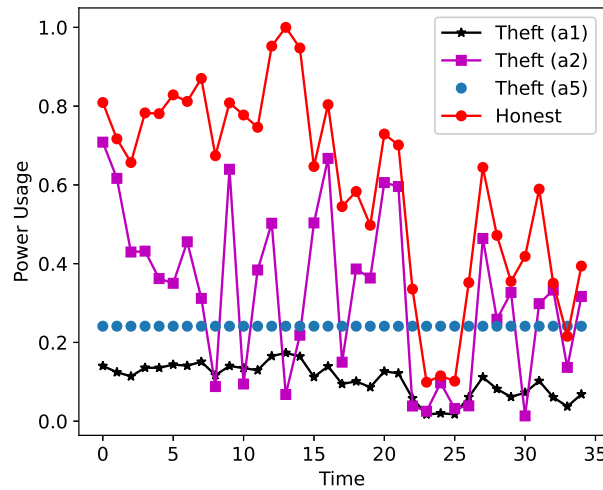


Figure 6. Attack patterns 1, 2, and 5.

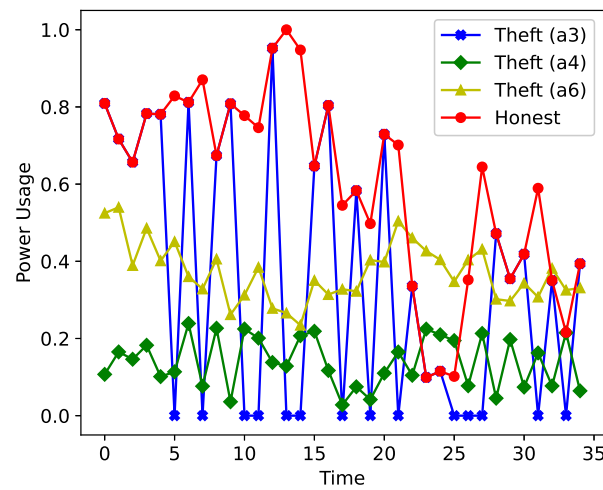


Figure 7. Attack patterns 3, 4, and 6.

Table 1. Representation of the shortcomings and the proposed solutions.

Shortcomings	Proposed Solutions	Evaluation
L1 and L2: imbalanced dataset issue and inadequate training data	S1: Employ six theft attacks on normal samples, then apply SMOTE to balance the dataset	V1: Comparison with oversampling techniques
L3: Misclassification as a result of non-malicious circumstances	S2: Integrate auxiliary data	V2: Performance comparison with traditional models
L4: Inappropriate tuning of model’s hyperparameters	S3: RSA	V3: Compare the RSA with the existing GRA approach

2.2.3. Gated Recurrent Unit Network for Smart Meter Data

The GRU is a variant of the LSTM that overcomes the computational complexity of the LSTM by considering few gates, as it eliminates the output gate. The GRU includes

an update gate (long-term memory) and a reset gate (short-term memory), as shown in Figure 8.

$$r_t = \sigma(X_t * V_r + H_{t-1} * W_r + B_r), \quad (1)$$

$$u_t = \sigma(X_t * V_u + H_{t-1} * W_u + B_u). \quad (2)$$

According to Equations (1) and (2) [15,28], r_t and u_t denote the number of times the reset gate and update gate have been enabled, respectively. V_r and V_u denote the weights of the input layer, while W_r and W_u indicate the recurrent weights of the GRU. The biases of the deep network are denoted by the variables B_r and B_u . X_t is the current input state and H_{t-1} is the previous layer input. All values of the reset and update gates are multiplied by the sigmoid activation function, denoted as σ [29].

$$D_n = c * d_1, c * d_2, c * d_3, \dots, c * d_i, (i = 1, 2, 3, \dots, 365). \quad (3)$$

Equation (3) indicates the daily energy consumption data over the year. $c * d_i$ presents the 365 days of consumption records. The GRU network examines the whole EC history of smart meters on a daily basis and generates the final result. The final predicted outcome of the GRU network and the output of the MLP network are activated using a single activation function to generate a combined prediction.

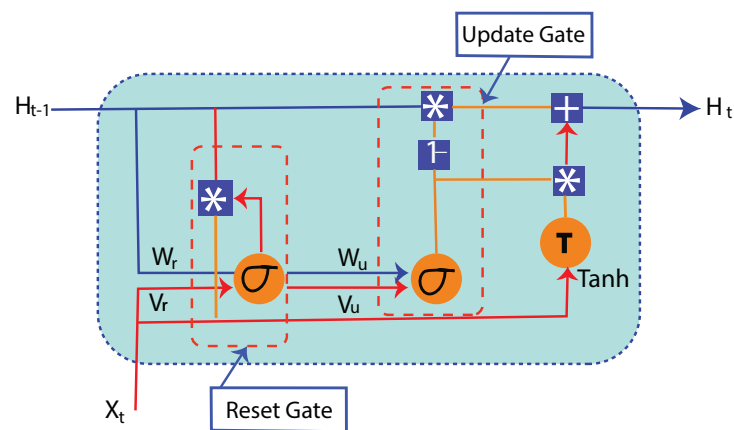


Figure 8. GRU model architecture.

2.2.4. Multi-Layered Perceptron Network with Auxiliary Data

The auxiliary dataset is analyzed using the MLP network. The MLP contains more than one hidden layer of neurons. The validation dataset is used to choose these hidden layers in the MLP network.

$$H_n = \sigma(\sum U_{i,n} * X_i + B_n), i = 1, 2, \dots, N, \quad (4)$$

$$Y_n = \sigma(U_n * H_{n-1} + B_n). \quad (5)$$

According to Equations (4) and (5) [4], U_n denotes the weights of layer n , H_{n-1} indicates the previous hidden states of the input layer, and B_n represents the bias. After processing the input values, the activation function that activates the neuron is called and determines whether to send the values to the next layer or not. σ represents the sigmoid activation function. Equation (5) shows the output layer, which is denoted as Y_n . In this study, we used the Rectified Linear Unit (ReLU) activation function in the hidden layer, while for the final output layer, a sigmoid activation function was used for the binary classification [15]. To accelerate the network convergence, a batch normalization layer was added to standardize the input values. Afterwards, a dropout layer was added as a regularization technique to reduce overfitting.

2.2.5. Random-Search-Based Parameters' Optimization Algorithm

A critical challenge in the development of deep learning models is the appropriate setting of hyperparameters to achieve optimal results. Inappropriate selection of hyperparameters adversely affects the training activity as well as the time complexity of deep learning models. The fundamental goal of deep learning classifiers is to improve the accuracy of the classification results. Therefore, the selection of a suitable learning rate, number of neurons, number of hidden layers, batch size, activation function, epochs, and other hyperparameters of deep learning classifiers has significant impacts on the model's performance. In order to achieve optimal results, the important hyperparameters of the classifiers need to be optimized (tuned). In recent research, the GSA has been considered in many machine and deep learning algorithms for the tuning of hyperparameters [4,30]. For instance, we used hyperparameters hp1, hp2 and hp3 of an ML model M. The GSA specifies the range of values for each hyperparameter. Afterwards, it creates many different M versions using different combinations of hyperparameter values. This range of hyperparameter values is referred to as the grid.

Moreover, the manual selection of hyperparameters and GSA makes it somewhat easier for the user to define these essential parameters. However, both techniques take a long time to converge. On the other hand, GSA remains a computationally intensive method, particularly as the number of hyperparameters grows and the interval between discrete values shrinks [30].

In the case of high dimensionality, when several hyperparameters drastically grow, the GSA method suffers a lot and is computationally overburdened. It takes the maximum time during tuning, even in cases with a small number of hyperparameters. Since there is no guarantee of finding the best solution, in this study, a RSA method was employed to improve the classification accuracy of the models. The RSA is a stochastic optimization algorithm that is invaluable for finding the optimal solution globally with fast-running simulations. It performs searching using random combinations of hyperparameter values to train a model. Additionally, it is more effective in high-dimensional space.

The RSA consists of five major steps:

- The initial value is stored in a variable, denoted by x .
- If the values stored in x are target node values, the algorithm immediately stops with the success. Otherwise, it moves to the next step.
- The values of x are updated to get the optimal possible combination of x . We obtain the number of child nodes (values of x) and store them in another variable C .
- A value from all possible combinations of child node values is randomly selected.
- The values of x are replaced with the new values, and then the process returns to step 2 for validation, where the existing values are compared with the target values. The process continues until the final optimal solution is reached. Figure 9 shows the process of tuning hyperparameters with the RSA.

To optimize the hyperparameters, we used the following steps.

Step 1: The hyperparameters are initialized with their possible range. To train our hybrid MLP-GRU model, the hyperparameters, such as the activation function, epochs, the number of hidden layers, batch size, etc., are defined. The RSA samples a set of values for each of these hyperparameters and makes a grid of all available values from their respective distributions and uses it for training.

Step 2: During the evolution period, only one solution is retained. A random vector is added to the solution after each epoch. The process is repeated numerous times, resulting in the training of several models.

Step 3: The new solution is checked after it has been measured. If the new solution is superior to the old one, then it is acknowledged as the correct one; otherwise, the old one remains unchanged.

Step 4: The best combination of the values of the hyperparameters is eventually preserved.

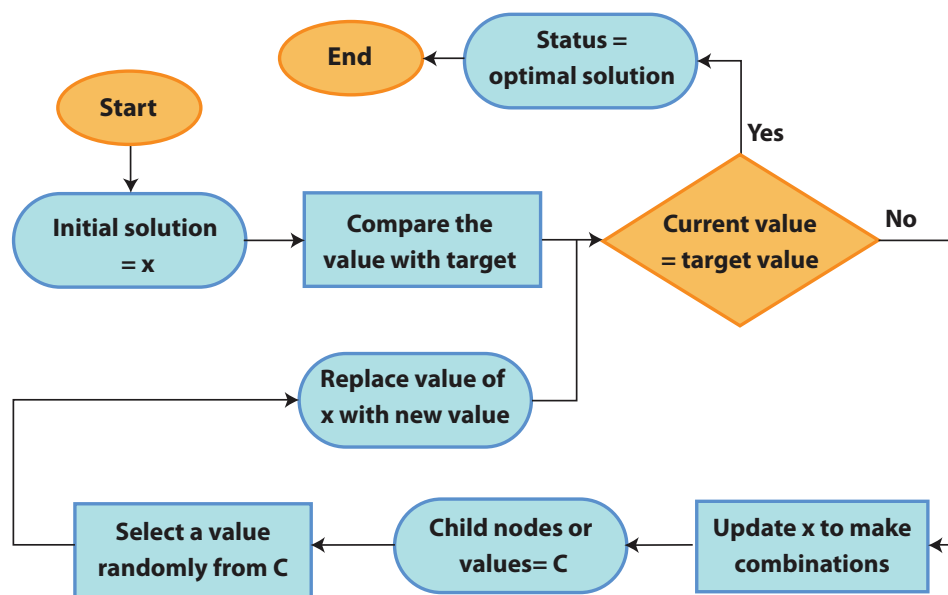


Figure 9. Flowchart of the Random Search Algorithm.

Table 2 displays the set of values that are searched and the best values that are discovered or revealed by the RSA during the tuning of the proposed MLP-GRU model. The best values are explored by tracking the results of the validation dataset. The RSA can monitor different random combinations of hyperparameters. To train the MLP-GRU model, the hyperparameters, like epochs, number of hidden layers, etc, are defined. The RSA can sample a set of values for epochs and hidden layers from their respective distributions that are used for training. The process is repeated several times until the desired results have been obtained. Table 3 shows the hyperparameters and their optimal values that were found using the GSA during the tuning of the existing MLP-LSTM model. However, we considered fewer hyperparameters due to their high computational time.

Table 2. Auxiliary Dataset Information.

Data Type	Description (MLP Input Data)	Size of Data
Residents' Information	Temporary residents and permanent residents	2
People	Total number of people including adults, children	3
Appliances	Number of appliances in a home including washing machine, fridge, iron, electronic devices, fans, AC, water-pump, UPS, water-dispenser, refrigerator and lightening devices	11
Connection Type	Single-phase and multi-phase	2
Rooms' Information	Number of rooms including bed room, living room, kitchen, washroom, dining room	6
Roof or Ceiling	The total height of ceiling, ceiling insulation used, ceiling insulation not used	2
Building Year	The year of building construction	1
Property Area	The area or location of house	1
Floors	The total number of floors in a building	1

Table 3. The Proposed MLP-GRU-Random Search Method.

Hyperparameter	Optimal Value	Values Range
Units	100	100, 10, 15, 50, 20, 35, 400, 25
Optimizer	Adam	Adam, Adamax and SGD
Dropout	0.01	0.3, 0.2, 0.5, 0.01, 0.1
Batch-size	32	10, 32, 25, 15
Activation function	relu	relu, elu, sigmoid, softmax, tanh and linear
Epochs	10	15, 25, 10, 20

3. Performance Measurement Indicators

In this section, we conduct a thorough examination of the proposed hybrid model's performance in comparison with the existing hybrid MLP-LSTM classifier. The accuracy, F1-score, PR-AUC, and ROC-AUC are effective performance indicators that are used to evaluate the performance of the techniques. These indicators are determined using the core confusion metrics, which are composed of four crucial error rates: False Positive (FP), False Negative (FN), True Positive (TP) and True Negative (TN) [31]. These metrics indicate the total number of consumers wrongly classified as thieves, accurately labeled as fair consumers, erroneously identified as honest consumers, and correctly labeled as thieves [32]. Accuracy is a widely used performance measure that represents the percentage of correct model predictions. It offers the measures of predictability for TPs and TNs in the classifier. It quantifies how well the model predicts TPs and TNs. However, it frequently fails for imbalanced datasets.

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}). \quad (6)$$

Mathematically, Equation (6) exhibits accuracy [31]. Other metrics were employed due to the lack of a specific measure for FP and FN predictions. The harmonic mean of recall and precision is called the F1-score, which is calculated by Equation (7).

$$\text{F1-score} = 2 \times (\text{Recall} \times \text{Precision}) / (\text{Recall} + \text{Precision}). \quad (7)$$

One of the main objectives of ETD is to enhance the TPR or Detection Rate (DR) while simultaneously reducing the FPR. Thus, the ROC-AUC is a useful metric for identifying NTLs in binary classification problems [33]. It demonstrates the relationship of TPR with FPR at different threshold values. A score is a number between 0 and 1 that represents how different the two classes are. An AUC score of 1 indicates a perfect detection method. In the case of an imbalanced dataset problem, it is more reliable in terms of evaluating the model's performance. FPR and TPR are beneficial for assessing the performance of a model. However, the precision of the model is not considered using these measures. Thus, the PR-AUC is a valuable performance measure that is used for evaluating the model's performance. It is more appropriate for imbalanced datasets as compared with balanced datasets.

4. Simulations and Findings

The simulation findings of the proposed model are discussed in this section. The PRECON dataset was used to test the proposed model. Python was used to carry out the simulations. The proposed model was implemented using an Intel Core i3 with 4 GB of RAM. Additionally, a Google Colaboratory application was used in conjunction with Python language packages such as NumPy, pandas, TensorFlow, Keras, etc. to simulate the data.

4.1. Data Acquisition

The proposed model was trained and tested on the actual smart meter dataset of PRECON [34], which is publicly available. It contains information about energy demand, including half-hourly electricity usage records from 42 residential properties. The PRECON smart meter dataset includes half-hourly energy consumption data with 48 features that belong to the normal users' consumption class, while the auxiliary dataset contains 28 features. Based on the dataset of honest users, we identified six types of theft attacks to generate malicious users' consumption patterns. We divided the consumption behavior of users on a half-hourly basis, while attacks were assessed on a yearly consumption basis. This helped us to analyze the daily consumption behavior of a user and identify nonmalicious users' patterns, as was done in [35,36] (to identify the periodicity in consumption). After applying attacks, we oversampled the minority class (benign) samples using SMOTE to balance the malicious attack class. The auxiliary information was provided to identify nonmalicious factors, which can cause a high misclassification rate. This included information on high energy consumption equipment and load profiles for the entire home. The utility provided the labeled dataset by inspecting it at least once. As a result, it is reasonable to assume that all samples belonged to trustworthy consumers. In addition, the dataset was split into a ratio of 80% training and 20% testing samples in a stratified manner, where 3494 instances were used in the training set and 874 were used in the testing set. Moreover, being motivated by [4], the proposed model was compared with the deep learning models and not the traditional machine learning models.

4.2. Evaluation Results

Figure 10 demonstrates the ROC curve of the proposed model after data balancing. The comprehensive scores of measurement are shown in Table 4. The proposed MLP-GRU model bet the single GRU classifier on the test data with an AUC score of 0.93. This indicates that the incorporation of auxiliary information such as permanent occupants, property area, and contracted power improve the performance by lowering the FPR. We also noticed that the performance of the hybrid MLP-LSTM was quite similar to our model, obtaining an AUC score of 0.89 due to the use of auxiliary information, except that the F1-score was relatively low with 0.89 compared with our model which obtained a score of 0.92. In contrast to our proposed model, the existing MLP-CNN classifier obtained an AUC score of 0.84 due to the lack of generalization in the CNN model.

Table 4. Hybrid MLP-LSTM-Grid Search.

Hyperparameter	Optimal Value	Values Range
Dropout	0.2	0.2, 0.5
Units	10	100, 10, 50
Optimizer	Adam	Adam and SGD
Activation function	sigmoid	relu and sigmoid

A comparative analysis of the proposed and existing models is depicted in Figure 10. On the x -axis and y -axis, the TPR and FPR are shown, respectively. The TPR indicates the proportion of correctly classified positive samples among all available data, whereas the FPR denotes the proportion of negative samples incorrectly classified as positive. The proposed model accurately categorized samples with high DR and low FPR values at the initial level. With a rising FPR, a small change was noticed after attaining a high TPR of 0.8. Hence, our proposed model's FPR was significantly lower than that of the existing model.

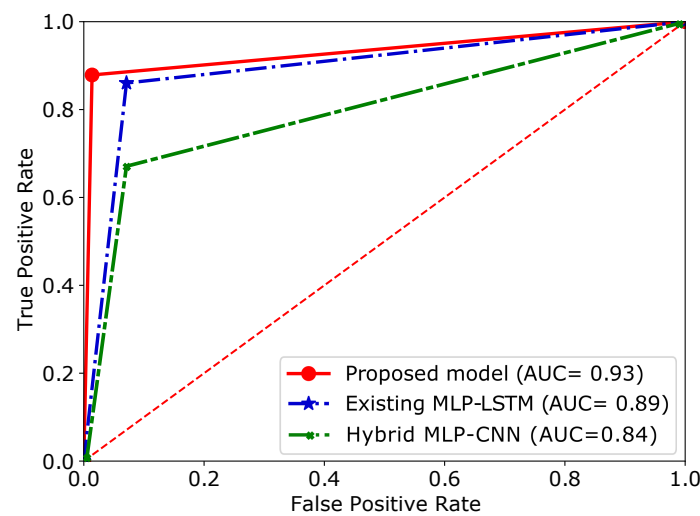


Figure 10. ROC-based performance comparison.

Subsequently, we were able to see an exponential periodic regain in the ROC curve of the proposed hybrid model. The models' stability and precision were enhanced by increasing the TPR and thereby reducing the FPR. A decrease in FPR minimizes the need for onsite inspections, which is a costly process as it involves the reliance on experts. Similarly, the PR-AUC curve is depicted in Figure 11. It indicates that, on the test datasets, our proposed model achieved a PR-AUC score of 0.91, which is substantially higher than those of existing models. Table 5 shows the accuracy, F1-score, and AUC values. The results indicate that the proposed MLP-GRU model surpasses the other state-of-the-art models. The computational complexity of the GRU classifier is minimal, since fewer gates are employed in GRU as compared with the LSTM classifier. In this regard, the GRU model requires a limited amount of hyperparameters for tuning, leading to a fast convergence rate. Moreover, we applied the RSA instead of the GSA, which is a computationally demanding process. The use of a small dataset also makes it better. In addition, the loss of the proposed model is shown in Figure 12. We ran 25 iterations. The loss declined with each move, settling at a 0-point minimum during training and testing. Training and testing data losses were the same. The proposed model works well regarding training and testing data, as seen in Figure 13.

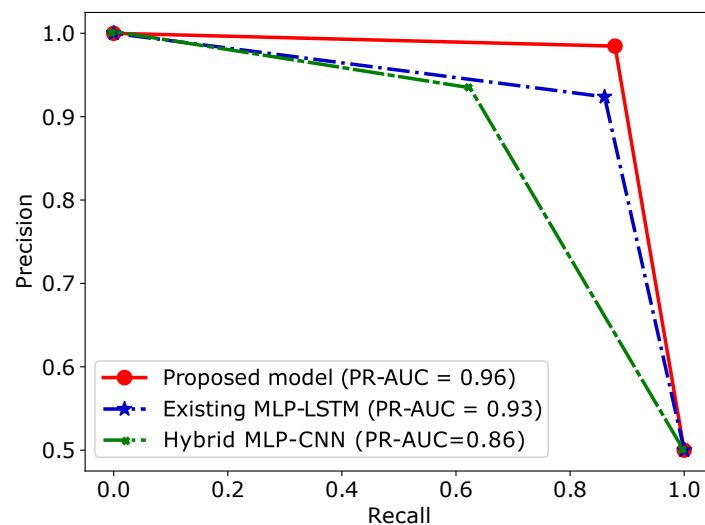


Figure 11. PR-AUC-based performance comparison.

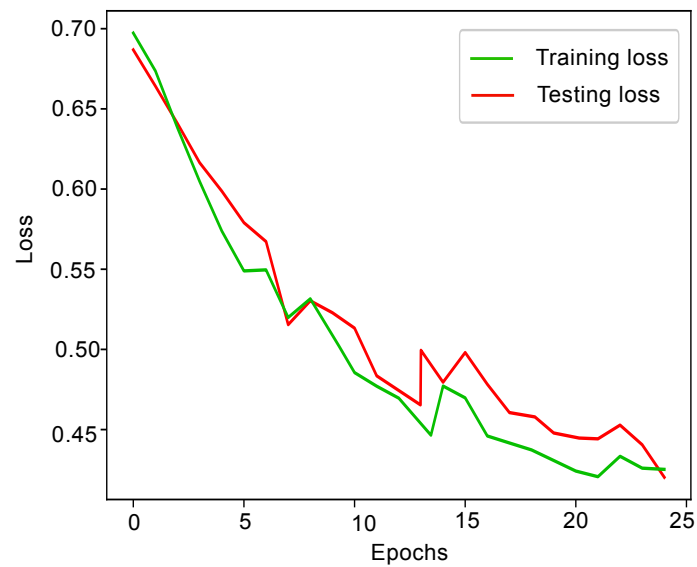


Figure 12. Accuracy-based analysis of the proposed solution.

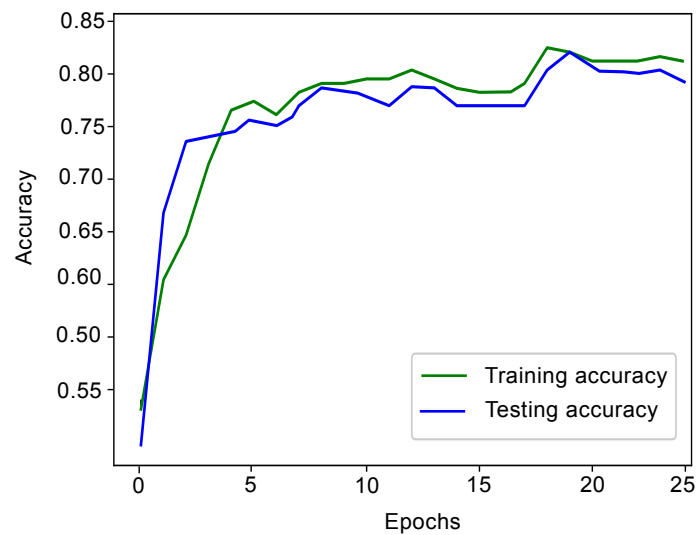


Figure 13. Loss-based analysis of the proposed solution.

It is critical to consider the execution time in a hyperparameter optimization process with real-world circumstances. Many studies have stated that finding acceptable hyperparameter values for a model can take a significant amount of time. As a result, many researchers do not consider parameter tuning due to the waste of time.

Figure 14 depicts the average execution times of the RSA and GSA, while Figure 15 shows the accuracy levels of the proposed and existing models. The results demonstrate that our proposed RSA approach takes less time than the existing MLP-LSTM classifier using the GSA method. The RSA creates a grid using a range of different hyperparameter values and picks random combinations from it to train the model. In contrast to the RSA, the GSA method makes a grid of hyperparameter values for each combination, which is computationally very expensive in terms of processing power and time. Besides, Table 5 compares the proposed model's performance with that of the existing models.

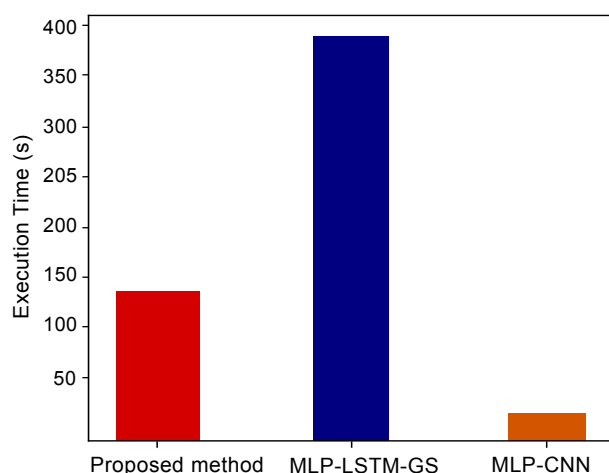


Figure 14. Execution time.

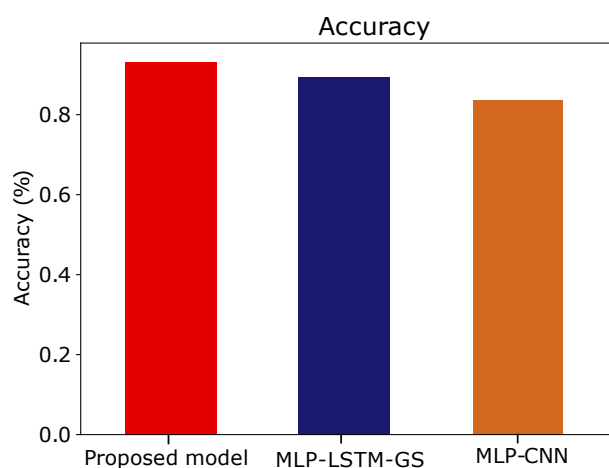


Figure 15. Accuracy analysis.

Table 5. Comparison of the proposed model's performance with that of existing models.

Models	Accuracy	AUC	F1-Score	Time Required (s)
Proposed model	0.93	0.93	0.92	144
MLP-LSTM-GS	0.89	0.89	0.89	391
MLP-CNN	0.67	0.84	0.71	26

5. Conclusions

In the proposed study, a hybrid deep learning model, MLP-GRU, was developed using metering data and auxiliary information. The MLP network receives auxiliary data, whereas the GRU network takes nonsequential or metering data for detecting electricity theft. Additionally, the EC datasets contain a small number of malicious samples that skew the model in favor of the majority class. The issue of the biased dataset is tackled through data augmentation in which synthetic theft instances are created via six different theft attacks on benign samples. Afterwards, the imbalanced dataset problem is resolved using SMOTE. The effectiveness of our proposed hybrid model was assessed against changes in EC usage patterns and various attack types. The PRECON dataset was used to run the simulations. According to the findings, the proposed model outperforms the existing hybrid MLP-LSTM and other conventional models. The results demonstrate that the proposed model performs much better with a ROC-AUC value of 0.93 and a PR-AUC value of 0.96 when auxiliary data are included with the metering data. In the future, there is a need to exploit more sophisticated optimization algorithms for adjusting

hyperparameters of deep learning classifiers to find the optimal ETD results. We will also use other residential areas of the PRECON dataset to conduct a detailed analysis of consumers' consumption behaviors.

Author Contributions: Writing—original draft preparation, B.K.; writing—review and editing, N.J. and E.A.M.; visualization, U.Q.; supervision, N.J.; project administration, A.A.; funding acquisition, project management, evaluating the prior works/literature review, and conducting a critical study, A.A. and N.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors thank King Saud University, Riyadh, Saudi Arabia for providing the funding for Project number (RSP-2021/295).

Data Availability Statement: The data is available at <http://web.lums.edu.pk/~eig/CXyzsMgyXGpW1sBo>, accessed on 15 September 2022.

Acknowledgments: The authors would like to acknowledge the support of Researchers Supporting Project number (RSP-2021/295), King Saud University, Riyadh, Saudi Arabia.

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

Abbreviations

The following abbreviations are used in this manuscript:

Abbreviation	Full Form
AMI	Advanced Metering Infrastructure
CNN	Convolutional Neural Network
CPBETD	Consumption Pattern Based Electricity Theft Detector
CVAE	Conditional Variational Auto Encoder
DNN	Deep Neural Network
DR	Detection Rate
EC	Electricity Consumption
ETD	Electricity Theft Detection
ETs	Extra Trees
FPR	False Positive Rate
GBTD	Gradient Boosting Theft Detector
GRU	Gated Recurrent Unit
GMM	Gaussian Mixture Model
GSA	Grid Search Algorithm
KNNs	K-Nearest Neighbors
LR	Logistic Regression
LSTM	Long-Short Term Memory
ML	Machine Learning
MLP	Multi-Layer Perceptron
NTL	Nontechnical Loss
PRECON	Pakistan Residential Electricity Consumption
RF	Random Forest
RSA	Random Search Algorithm
SVM	Support Vector Machine
SMOTE	Synthetic Minority Oversampling Technique
SSDAE	Stacked Sparse Denoising Auto-Encoder
SETS	Smart Energy Theft System
TL	Technical Loss
TPR	True Positive Rate
WGAN	Wasserstein Generative Adversarial Network
XGBoost	Extreme Gradient Boosting
H_{t-1}	Previous Layer Input
r	Reset Gate
t	Time Period
u	Update Gate
X_t	Current Input State

References

1. Praju, T.K.S.; Samal, S.; Saravanakumar, R.; Yaseen, S.M.; Nandal, R.; Dhablya, D. Advanced metering infrastructure for low voltage distribution system in smart grid based monitoring applications. *Sustain. Comput. Inform. Syst.* **2022**, *35*, 100691.
2. Otuoze, A.O.; Mustafa, M.W.; Abioye, A.E.; Sultana, U.; Usman, A.M.; Ibrahim, O.; Abu-Saeed, A. A rule-based model for electricity theft prevention in advanced metering infrastructure. *J. Electr. Syst. Inf. Technol.* **2022**, *9*, 2. [[CrossRef](#)]
3. Fei, K.; Li, Q.; Zhu, C. Non-technical losses detection using missing values' pattern and neural architecture search. *Int. J. Electr. Power Energy Syst.* **2022**, *134*, 107410. [[CrossRef](#)]
4. 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**, *35*, 1254–1263. [[CrossRef](#)]
5. Kocaman, B.; Tümen, V. Detection of electricity theft using data processing and LSTM method in distribution systems. *Sādhanā* **2020**, *45*, 286. [[CrossRef](#)]
6. Aslam, Z.; Javaid, N.; Ahmad, A.; Ahmed, A.; Gulfam, S.M. A combined deep learning and ensemble learning methodology to avoid electricity theft in smart grids. *Energies* **2020**, *13*, 5599.
7. Ghorri, K.M.; Abbasi, R.A.; Awais, M.; Imran, M.; Ullah, A.; Szathmary, L. Performance analysis of different types of machine learning classifiers for non-technical loss detection. *IEEE Access* **2019**, *8*, 16033–16048. [[CrossRef](#)]
8. Shehzad, F.; Javaid, N.; Aslam, S.; Javaid, M.U. Electricity theft detection using big data and genetic algorithm in electric power systems. *Electr. Power Syst. Res.* **2022**, *209*, 107975. [[CrossRef](#)]
9. Pamir Javaid, N.; Javaid, S.; Asif, M.; Javed, M.U.; Yahaya, A.S.; Aslam, S. Synthetic Theft Attacks and Long Short Term Memory-Based Preprocessing for Electricity Theft Detection Using Gated Recurrent Unit. *Energies* **2022**, *15*, 2778. [[CrossRef](#)]
10. 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](#)]
11. Saeed, M.S.; Mustafa, M.W.; Sheikh, U.U.; Jumani, T.A.; Mirjat, N.H. Ensemble bagged tree based classification for reducing non-technical losses in multian electric power company of Pakistan. *Electronics* **2019**, *8*, 860. [[CrossRef](#)]
12. Gunturi, S.K.; Sarkar, D. Ensemble machine learning models for the detection of energy theft. *Electr. Power Syst. Res.* **2020**, *192*, 106904. [[CrossRef](#)]
13. Fenza, G.; Gallo, M.; Loia, V. Drift-aware methodology for anomaly detection in smart grid. *IEEE Access* **2019**, *7*, 9645–9657. [[CrossRef](#)]
14. Qu, Z.; Li, H.; Wang, Y.; Zhang, J.; Abu-Siada, A.; Yao, Y. Detection of electricity theft behavior based on improved synthetic minority oversampling technique and random forest classifier. *Energies* **2020**, *13*, 2039. [[CrossRef](#)]
15. Li, W.; Logenthiran, T.; Phan, V.T.; Woo, W.L. A novel smart energy theft system (SETS) for IoT-based smart home. *IEEE Internet Things J.* **2019**, *6*, 5531–5539. [[CrossRef](#)]
16. Yang, R.; Qin, X.; Liu, W.; Huang, Z.; Shi, Y.; Pang, Z.; Zhang, Y.; Li, J.; Wang, T. A Physics-Constrained Data-Driven Workflow for Predicting Coalbed Methane Well Production Using Artificial Neural Network. *SPE J.* **2022**, *27*, 1531–1552. [[CrossRef](#)]
17. Yan, Z.; Wen, H. Electricity theft detection base on extreme gradient boosting in AMI. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 2504909. [[CrossRef](#)]
18. Punmiya, R.; Choe, S. Energy theft detection using gradient boosting theft detector with feature engineering-based preprocessing. *IEEE Trans. Smart Grid* **2019**, *10*, 2326–2329. [[CrossRef](#)]
19. Huang, Y.; Xu, Q. Electricity theft detection based on stacked sparse denoising autoencoder. *Int. J. Electr. Power Energy Syst.* **2021**, *125*, 106448. [[CrossRef](#)]
20. Gong, X.; Tang, B.; Zhu, R.; Liao, W.; Song, L. Data Augmentation for Electricity Theft Detection Using Conditional Variational Auto-Encoder. *Energies* **2020**, *13*, 4291. [[CrossRef](#)]
21. Park, C.H.; Kim, T. Energy Theft Detection in Advanced Metering Infrastructure Based on Anomaly Pattern Detection. *Energies* **2020**, *13*, 3832. [[CrossRef](#)]
22. Jokar, P.; Arianpoo, N.; Leung, V.C. Electricity theft detection in AMI using customers' consumption patterns. *IEEE Trans. Smart Grid* **2015**, *7*, 216–226. [[CrossRef](#)]
23. Maamar, A.; Benahmed, K. A hybrid model for anomalies detection in AMI system combining K-means clustering and deep neural network. *Comput. Mater. Continua* **2019**, *60*, 15–39. [[CrossRef](#)]
24. 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](#)]
25. 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](#)]
26. Kabir, B.; Ullah, A.; Munawar, S.; Asif, M.; Javaid, N. Detection of non-technical losses using MLPGRU based neural network to secure smart grids. In Proceedings of the 15th International Conference on Complex, Intelligent and Software Intensive System (CISIS), Asan, Republic of Korea, 1–3 July 2021; ISBN 978-3-030-50454-0.
27. Kong, X.; Zhao, X.; Liu, C.; Li, Q.; Dong, D.; Li, Y. Electricity theft detection in low-voltage stations based on similarity measure and DT-KSVM. *Int. J. Electr. Power Energy Syst.* **2021**, *125*, 106544.
28. 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.

29. Kuo, P. H.; Huang, C.J. An electricity price forecasting model by hybrid structured deep neural networks. *Sustainability* **2018**, *10*, 1280. [[CrossRef](#)]
30. George, S.; Sumathi, B. Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 173–178.
31. Ghori, K.M.; Imran, M.; Nawaz, A.; Abbasi, R.A.; Ullah, A.; Szathmary, L. Performance analysis of machine learning classifiers for non-technical loss detection. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *18*, 1–16. [[CrossRef](#)]
32. Razavi, R.; Gharipour, A.; Fleury, M.; Akpan, I.J. A practical feature-engineering framework for electricity theft detection in smart grids. *Appl. Energy* **2019**, *238*, 481–494. [[CrossRef](#)]
33. Aslam, Z.; Ahmed, F.; Almogren, A.; Shafiq, M.; Zuair, M.; Javaid, N. An attention guided semi-supervised learning mechanism to detect electricity frauds in the distribution systems. *IEEE Access* **2020**, *8*, 221767–221782. [[CrossRef](#)]
34. Nadeem, A.; Arshad, N. PRECON: Pakistan residential electricity consumption dataset. In Proceedings of the Tenth ACM International Conference on Future Energy Systems, Phoenix, AZ, USA, 25–28 June 2019; pp. 52–57. Available online: <http://web.lums.edu.pk/~eig/CXyzsMgyXGpW1sBo> (accessed on 15 September 2022).
35. 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](#)]
36. Arif, A.; Javaid, N.; Aldegheishem, A.; Alrajeh, N. Big data analytics for identifying electricity theft using machine learning approaches in micro grids for smart communities. *Concurr. Comput. Pract. Exp.* **2021**, *33*, e6316. [[CrossRef](#)]