



ORIGINAL RESEARCH

Using machine learning ensemble method for detection of energy theft in smart meters

Asif Iqbal Kawoosa¹ | Deepak Prashar² | Muhammad Faheem³  | Nishant Jha²  |
Arfat Ahmad Khan⁴

¹School of Computer Applications, Lovely Professional University, Phagwara, Punjab, India

²School of Computer Science & Engineering, Lovely Professional University, Phagwara, Punjab, India

³School of Technology and Innovations, University of Vaasa, Vaasa, Finland

⁴Department of Computer Science, College of Computing, Khon Kaen University, Khon Kaen, Thailand

Correspondence

Muhammad Faheem, School of Technology and Innovations, University of Vaasa, 65200 Vaasa, Finland.

Email: muhammad.fatheem@uwasa.fi

Abstract

Electricity theft is a primary concern for utility providers, as it leads to substantial financial losses. To address the issue, a novel extreme gradient boosting (XGBoost)-based model utilizing the consumers' electricity consumption patterns for analysis is proposed for electricity theft detection (ETD). To remove the imbalance in the real-world electricity consumption dataset and ensure an even distribution of theft and non-theft data instances, six different artificially created theft attacks were used. Moreover, the utilization of the XGBoost algorithm for classification, especially to identify malicious instances of electricity theft, yielded commendable accuracy rates and a minimal occurrence of false positives. The proposed model identifies electricity theft specific to the regions, utilizing electricity consumption parameters, and other variables as input features. The authors' model outperformed existing benchmarks like k-neural networks, light gradient boost, random forest, support vector machine, decision tree, and AdaBoost. The simulation results using the false attacks for balancing the dataset have improved the proposed model's performance, achieving a precision, recall, and F1-score of 96%, 95%, and 95%, respectively. The results of the detection rate and the false positive rate (FPR) of the proposed XGBoost-based detection model have achieved 96% and 3%, respectively.

1 | INTRODUCTION

Electricity theft in emerging economies is a cause for concern. It is a severe problem today as substantial financial losses occur due to electricity stolen by illegal consumers without getting billed. Energy theft creates an imbalance in demand and supply to a great extent. In India, non-technical loss (NTL) due to electricity theft is calculated annually at approximately \$17 billion, which is 30% to 40% of total electricity generation. Globally, the annual monetary loss due to electricity theft is around \$96 billion [1]. Electricity theft is stopping the utilities to improve power networks and develop financially.

Electricity theft is a challenge for utilities. The existing theft detection methods using machine learning methods to detect various theft attacks are not efficient and have high positive rates [2]. To work machine learning methods efficiently and accurately on real-time data, ensemble techniques are gaining

its existence in electricity theft detection (ETD). Ensemble learning systems (ELS) are used more often than a single machine learning algorithm as ensemble models enhance the accuracy and make the final model more efficient and accurate.

The goal of this article is to propose an ETD model that can detect electricity theft efficiently by employing extreme gradient boosting (XGBoost) ensemble model for the classification of genuine and fraud users on the basis of their historical usage pattern, seasonal impact, and power scenario during different seasons (summers and winters) of the region under study. The analysis is done on the weekly, monthly, and seasonal basis to check the trend of electricity usage. The study also takes care of the unscheduled power cuts adapted in the region to compensate for gap in supply and demand.

- I. It is a challenge to train a machine-learning model on imbalanced dataset as the samples of theft instances are

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2023 The Authors. *IET Generation, Transmission & Distribution* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

much smaller in number than the non-theft instances. In order to balance the data samples, six artificial theft attacks were created and used in simulations to train the model. The balancing of theft and non-theft instances in an augmented dataset lowers the rate of false positives in the evaluation of performance. Our model performed well despite the change in the electricity usage pattern caused by seasonal changes. The model uses features from weather data and geographic data sources apart from the electricity consumption dataset. Overall the factors that were taken into consideration include electricity usage behaviour, temperature, vacations, weekends, etc., that may impact the usage pattern of the consumer. This study has made several important contributions, summarized as follows:

- II. In this study, we propose the development of a highly efficient XGBoost-based ensemble model for the purpose of detecting electricity theft attacks in smart meters. Smart meters record electricity usage of each customer at half-hour intervals for a duration of 1036 consecutive days. This implies that every consumer possesses 1036 vectors comprising 48 distinct values in 24 h. The focus of this approach is on improving the detection rate (DR) and lowering the false positive rate (FPR) in ETD by tailoring the input features for enhancing the effectiveness, accuracy of the model. The proposed XGBoost-based detection model utilizes the load profile of the customer's energy consumption and adds features from auxiliary datasets to acquire knowledge about unique patterns of electricity usage. This knowledge allows the detector to effectively differentiate between honest and malicious energy consumption values.
- III. After using forward fill, three-sigma and z_score methods for filling in the missing values and removal of outliers and standardization of data values respectively. False theft data samples are synthetically constructed to augment the dataset to balance the dataset. The model is trained on the training set both on imbalanced and balanced datasets in a ratio of 80% and 20% of the samples, respectively. The obtained results are compared with the test set. The Xgboost-based detection model is selected based on performance. Grid search analysis is conducted to optimize the hyper-parameters of the XGBoost model. The experimental findings demonstrate that the XGBoost ensemble machine-learning model.

2 | RELATED WORK

In the realm of the ETD in recent years, the literature available has been classified into three distinct methodologies [3]: (1) State estimation based [4], (2) Game theory-based [5], and (3) Artificial Intelligence-based machine learning algorithms [6].

2.1 | State-estimation based

In the previous studies [3–6], researchers have explored the utilization of specific external hardware devices and designs,

specialized metering devices, distribution transformers, sensors, and various types of metering devices, for ETD. A specific method discussed in [7] involves the utilization of an adapted ammeter device for theft detection on the low-voltage (LV) side of the power network. This approach focuses on comparing the differences in electrical parameters between local and remote devices to identify theft [7]. The state estimation method is employed at the substation level to detect anomalies and, subsequently, electricity theft within a cluster [7]. However, this method has certain drawbacks, primarily the high cost associated with the implementation of additional devices and the inherent challenges involved in installing these devices within the existing system. The maintenance cost of this method of detecting electricity theft is very high. This approach is commonly referred to as the network-oriented ETD method.

2.2 | Game theory-based method

This methodology is grounded in the principles of game theory, which involves the strategic interactions between dishonest consumers (electricity thieves) and utilities [8]. In the game theory, the objective is to attain a Nash equilibrium, whereby the actions of a dishonest consumer in attempting to steal electricity are deterred [9, 10]. Employing the game theory approach presents certain advantages in terms of cost-effectiveness, albeit it poses challenges in establishing precise functions for each customer and the utility company for theft detection [8, 9].

2.3 | Hybrid method

A hybrid method integrates network-oriented estimation, traditionally used for theft detection at the medium voltage level (sub-station level), and artificial intelligence techniques applied at the low voltage or distribution level (consumer end) [2]. This estimation technique utilizes network-oriented measurements, such as power flow and voltage measurements, to estimate the state variables of the power system. By analyzing deviations from the expected states, indicators of potential theft can be identified. At the low voltage level or distribution level, machine learning techniques are employed. These techniques leverage historical data and relevant features to train models capable of detecting theft patterns based on consumer behaviour and consumption patterns [2]. By combining the outputs of the state-based estimation and machine learning models, the hybrid method achieves comprehensive and accurate theft detection.

2.4 | Data-oriented method

While hybrid methods incur additional hardware costs, the utilization of machine learning techniques for ETD has become widespread among utilities due to their feasibility. However, many existing machine learning-based ETD techniques struggle to reduce the FPR and enhance the true DR. Researchers are actively striving to improve these performance metrics for

increased efficiency. By leveraging machine learning methods, utilities can leverage extensive historical data to identify the trend in electricity usage, thereby analyzing consumer behaviour effectively. Despite the integration of Advanced Metering Infrastructures (AMIs) like sensor devices, Internet of Things (IoT), and smart meters, the existing energy theft detection methods have not completely eradicated electricity theft [10]. The implementation of smart meters enables the remote sharing of consumers' electrical energy usage.

This research paper builds upon existing literature [6, 11, 12] and proposes a novel model for detecting electricity theft. The model utilizes consumer consumption behaviour to identify anomalies in current usage patterns. The XGBoost-based detector is utilized which consists of two phases. During the training phase, normal metering data is appropriately formatted and subjected to data pre-processing procedures, which encompass addressing any missing or erroneous values and performing normalization. The dataset exhibits a smaller number of theft or malicious samples, which is consistent with typical datasets of this nature. The malicious samples were generated by modifying benign samples based on six different attack types [13]. The XGBoost-based detection model is then trained on both benign and malicious samples. In the application phase, the trained model is used to classify new samples into benign or malicious classes on test data. The simulations are conducted using the State Grid of China Corporation (SGCC). Compared to other algorithms like LightGBM, CatBoost, AdaBoost, SVM, AR, BN, NN, LR, DT, and RF, used in several research papers [14–20], the proposed method shows higher accuracy and lower FPR in ETD. The experiment results highlight the excellent performance of the proposed method, especially when dealing with the external factors impacting the usage pattern.

The rest of the paper is organized as: Section 3. Methodology proposals, Section 4. Model comparison and selection, Section 5. Model Selection, Section 6. Model Evaluation, Section 7. Discussions and Comparison, Section 8. Conclusion, Section 9. Future research.

3 | METHODOLOGY PROPOSALS

The methodology of this research involves the use of an XGBoost-based detector for ETD by utilizing the historical consumption patterns of the consumers along with other relevant features. The raw dataset as used in [21] of the State Grid Corporation of China (SGCC) is preprocessed before being put into use. The preprocessing steps include filling in missing values with the forward fill method, removing outliers using the three-sigma rule (TSR), and standardizing the data using the Min–Max method. To balance the theft and non-theft instances in the dataset, six distinct theft attacks are artificially constructed, similar to the real theft instances present in the dataset. Only those theft instances that fall within the Interquartile Range (IQR) of the theft data are used. (More in Section 3.4)

For training and testing machine learning models, both unbalanced and balanced datasets are utilized. Principal Component Analysis (PCA) is employed to reduce the dataset's dimensions,

and various parameters of electricity usage are included, along with features extracted from statistical techniques and auxiliary databases such as the weather database, customer tariff details, customer's previous records, electricity consumption trends, seasonal impacts on electricity usage, usage variations during weekdays, weekends, holidays, vacations, electricity sub-station curtailment schedules, and more.

Multiple models are trained on both the unprocessed and balanced datasets, with XGBoost emerging as the most accurate, demonstrating high DR rates and low FPRs for electricity theft. The XGBoost-based detector serves the purpose of classifying the data points into theft and non-theft instances, utilizing the discernible patterns in electricity consumption as shown in Table 1. Data aggregation plays a crucial role in facilitating the model's ability to discern patterns in consumer usage and extract implicit information. The proposed XGBoost-based model, as shown in Figure 1, utilizes machine learning techniques, shows a high level of accuracy and scalability in detecting instances of electricity theft. The proposed model exhibits superior performance compared to the widely recognized energy theft detection models. See Figure 1 in Table 4. The State Grid Corporation of China SGCC dataset consists of the vast consumer data of electricity consumption for analyzing electricity usage behaviour and for the comprehensive simulation of diverse forms of electricity theft attacks. The objective of this study is to identify and mitigate instances of energy theft in order to enhance the stability and efficiency of the national power grid. The achievement of this objective is demonstrated by the utilization of XGBoost ensemble classifiers on smart meter data.

3.1 | Dataset details

The dataset of SGCC is used in this study to cover the vast consumer base for analyzing electricity usage behaviour. The SGCC dataset consists of a total of 42,372 records, with 3615 instances representing abnormal consumer data and 38,757 instances representing normal consumer data. The data is collected at 30-min intervals throughout the specified time as indicated in Table 2. The proposed model is executed on the available data w.e.f. 2014 to 2016 (SGCC).

The dataset comprises various attributes about electricity consumption, power parameters, and consumers' profile information. These attributes encompass details such as tariff agreement, type of residential house, list of registered gadgets, count of persons living in the house, and occupation of other family members [21]. In addition, the weather conditions during the analyzed period are taken into account to ensure precise identification of the electricity theft, taking into consideration the adverse temperatures and power availability.

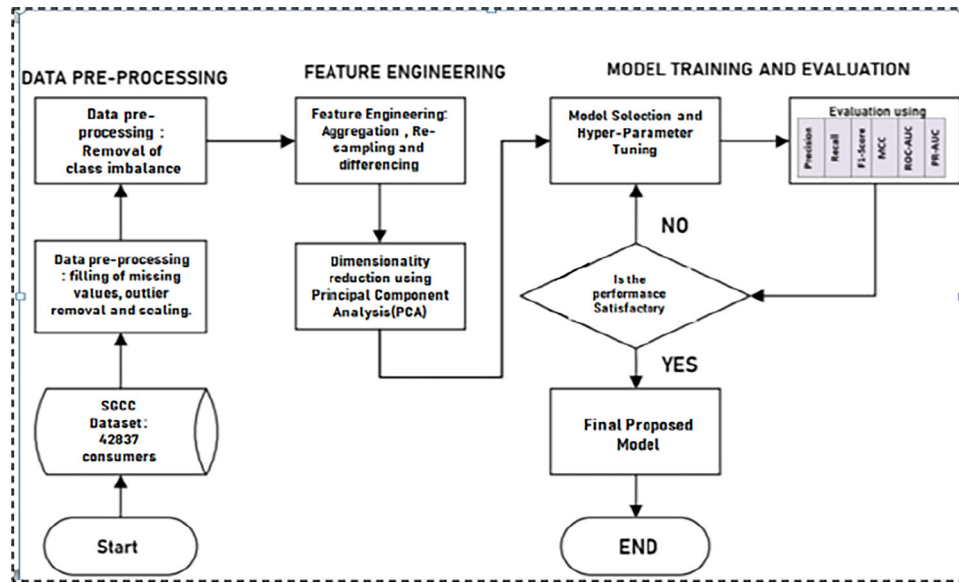
3.2 | Filling of missing values

The SGCC dataset contained missing values probably due to meter failure or unreliable transfer of data on the network, or

TABLE 1 Mapping of problems identified, solutions proposed, and validation.

Problem	Proposed solution	Validation
Imbalance in dataset	Balanced by adding false theft instances	Generates six false theft attack closer to real theft
Feature extraction	Statistical functions used for training the model on patterns	Generates usage patterns
Classification of theft and normal instances	XGBoost-based ensemble model	Better results than other state-of-the-art ETD models

ETD, electricity theft detection; XGBoost, extreme gradient boosting.

**FIGURE 1** Framework of proposed model for the detection of electricity theft using SGCC dataset. SGCC, State Grid of China Corporation.**TABLE 2** Details of SGCC dataset.

Total time of study	January 2014 to October 2016
Total consumers	42,372
Electricity stealers	3615
Genuine consumers	38,757

SGCC, State Grid Corporation of China.

due to the storage issues. The missing data is filled using the forward filling method to fill in the fields having NaN values. The techniques like forward filling, backward filling, linear interpolation, mean of nearest neighbours etc., are considered viable for filling the time-series data instead of using medians or means. So the forward fill method was employed here to fill in values as mentioned below:

$$\text{dat}_{a_{\text{freq}}} = \text{data.asfreq}('D', \text{method} = 'ffill') \quad (1)$$

The utilization of forward filling is justified due to the proximity of the intervals and the assumption that if data is absent for a given interval, it can be presumed to be the same as in the preceding interval. The data collection frequency of smart meters is typically at 15-min intervals. However, for the

purposes of our study, we adjusted the observation interval to half-hourly intervals due to the absence of significant changes. This modification resulted in a total of 48 data points per day. The dataset was found to be incomplete in terms of frequency information, as evidenced by the absence of a frequency value (freq = 'None'). The frequency was modified to take place at intervals of 30 min. Pandas provides a range of frequency options for calculating frequencies, such as hourly ('H'), daily ('D'), weekly ('W'), monthly ('M'), annual ('A'), and additional options. Nevertheless, the current system lacks provisions for intervals of 30 min or half-hourly frequencies. To generate frequencies at half-hour intervals, the date_range function was employed to create a DatetimeIndex at half-hourly intervals by specifying a frequency of '30 min', as shown below:

$$\begin{aligned} \text{half_hourly_range} &= \text{pd.date_range}(\text{start} = \text{start_date}, \\ &\text{end} = \text{end_date}, \text{freq} = '30\text{min}') \end{aligned}$$

The electricity consumption data may have erroneous values recorded by the energy meter malfunctioning. They are treated as outliers and are removed using the 'TSR of thumb'. Outliers are removed using the TSR rule as mentioned in [10]

TABLE 3 Different types of artificial theft attacks.

Attack types	Modifications	Remarks
Type 1	$\tilde{x}_i = \alpha x_i$	$0.2 < \alpha < 0.8$
Type 2	$\tilde{x}_i = \alpha_t x_i$	$0.2 < \alpha_t < 0.8$
Type 3	$\tilde{x}_i = \beta x_i$	$\beta = 1$ if $36 > t > 20$ $\beta = 0.5$ otherwise.
Type 4	$\tilde{x}_i = \alpha_t \bar{x}$	$0.2 < \alpha_t < 0.8$
Type 5	$\tilde{x}_i = \bar{x}$	
Type 6	$\tilde{x}_i = x_{48-t}$	

$$f(v) = \begin{cases} \frac{N}{2}, v_i \in NaN, v_i(m-1), v_i(m+1) \in NaN \\ 0, v_i \in NaN, v_i(m-1) \text{ or } v_i(m+1) \in NaN \end{cases}$$

$$\forall v_i \text{ and } v_i \notin NaN \quad (2)$$

$$O(v_i, t) = \text{wif } v_i(\bar{t}) > v_i(\bar{t}) \text{ otherwise}$$

where

$$w = \text{avg}(v_i) + 2s(v_i(t)) \quad (3)$$

After filling in the missing and erroneous values and removing outlier values as also done in [10], the data values are normalized using min–max normalization.

$$N(v_i(t)) = \frac{v_i(t) - \min(v)}{\text{imax}(v) - \min(v)} \quad (4)$$

$v_i(t)$ is the usage of electricity at time t [10], $\min(v)$ is the usage of minimum electricity [10], and $\text{imax}(v)$ is the usage of electricity at the time (t) .

In the anomaly detection process, it is important to analyze how electricity is used by fraudulent and normal users. This paper also uses six types of fake theft attacks to balance the dataset.

3.3 | Removal of the class imbalance by injecting synthetic theft attacks

In scenarios characterized by significant imbalances, such as datasets pertaining to electricity consumption that include huge imbalances between theft and non-theft instances, XGBoost detector may too exhibit a tendency to prioritize the majority class [22]. The theft attacks are generated synthetically. We assume that no fraudulent users have altered any of the historical data. The consumers' daily metering data are denoted by the notation $x = (x_1, x_2, x_{48})$ (reading after every 30 min in 24 h). Smart meters communicate metering data (in Kilowatts) to the data management system every 30 min as depicted in Figure 2. We employ the techniques suggested in [23] to synthetically generate six attack types to alter metering data and produce malicious samples. Table 3 outlines the specifics of how

1	smkpdd_id	timestmp	energy Kw/
2	MAC000002	2014-10-12 00:30:00.0000000	28356
3	MAC000002	2014-10-12 01:00:00.0000000	28356.92
4	MAC000002	2014-10-12 01:30:00.0000000	28357.4
5	MAC000002	2014-10-12 02:00:00.0000000	28359.7
6	MAC000002	2014-10-12 02:30:00.0000000	28360.12
7	MAC000002	2014-10-12 03:00:00.0000000	28361.08
8	MAC000002	2014-10-12 03:30:00.0000000	28364.26
9	MAC000002	2014-10-12 04:00:00.0000000	28365.53
10	MAC000002	2014-10-12 04:30:00.0000000	28366.97
11	MAC000002	2014-10-12 05:00:00.0000000	28367.8
12	MAC000002	2014-10-12 05:30:00.0000000	28368.95
13	MAC000002	2014-10-12 06:00:00.0000000	28370
14	MAC000002	2014-10-12 06:30:00.0000000	28371.04
15	MAC000002	2014-10-12 07:00:00.0000000	28373.12
16	MAC000002	2014-10-12 07:30:00.0000000	28375.73
17	MAC000002	2014-10-12 08:00:00.0000000	28379.13
18	MAC000002	2014-10-12 08:30:00.0000000	28379.64
19	MAC000002	2014-10-12 09:00:00.0000000	28382.55
20	MAC000002	2014-10-12 09:30:00.0000000	28385.84

FIGURE 2 Dataset instance of a single meterID.

artificial theft attacks are developed in this study also used in [23].

Type 1 defines an attack where a smart meter's reading is multiplied by the same parameter (say αt) all the time during a day (where αt ranges from 0.2 to 0.8).

Type 2 defines an attack where a smart meter's reading is multiplied by a different random number at different times (say αt).

Type 3 attack is defined as such when a smart meter sends readings of 0.5 (half of the actual load) during the peak load time, that is, night hours during the winter, and actual load during off-peak hours. The meter is in a compromised state between the hours of 6:00 pm till 10:00 am the next day. During the daytime or inspection time, the meter works in a normal state. The 0.5 of the actual load matches the average load of a normal consumer in a cluster or neighbourhood, making it difficult for the machine learning algorithm to detect any anomaly or electricity theft unless the observed load crosses the predicted load threshold.

Type 4 is defined as that form of theft attack that sends the average value multiplied by a random factor (αt) of energy consumption reading to the utility management system.

Type 5 is defined as a form of energy theft attack that indicates that energy meters send the average value of energy consumption during the day to the control centre of the utility.

Type 6 is defined as that form where the fraudulent consumers send only reverse the order of consumption reading during the day to avoid high dynamic pricing during peak hour's period.

All six attacks are tested separately as well as in combination to test the performance of the proposed XGBoost-based model taking into consideration the weather and erratic power supply conditions as depicted in Figure 3. The proposed model was checked on precision, recall, FPR, and AUC of the proposed model in detecting all the attacks. From the test results, it can be found that our method has an excellent performance in detecting all attack types except for an attack of type 2. This attack type does not allow a machine learning algorithm to detect an anomaly. Our approach uses extra features extracted from auxiliary databases to reduce FPR considering the erratic power

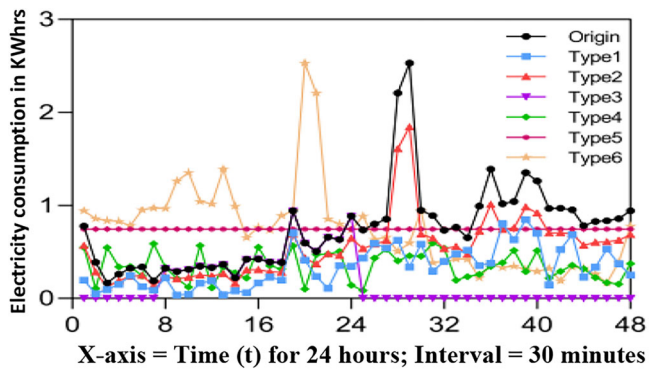


FIGURE 3 Shows an example of electricity used on a normal day, as well as six types of attacks (time interval = 30 min).

supply in the cluster or segment under consideration due to a huge gap in demand and supply available, a cluster under load-shedding due to overload in the system, or an area having inclement weather conditions. The learning of our proposed XGBoost-based ETD model is fine-tuned by selecting the subset of features extracted from EC and auxiliary datasets. Here the attacker does complex ways to steal electricity and does not follow a normal pattern challenging a machine learning model on accuracy. Gradient-boosting models like XGBoost are widely recognized for their robustness and ability to handle class imbalances. The model may still exhibit a tendency to prioritize the majority class. The theft data generated are selected by using the IQR to select attacks that are closer to real-world electricity theft. The IQR is a measure of statistical dispersion that represents the spread of the middle 50% of the data. It is calculated as the difference between the third quartile ($Q3$) and the first quartile ($Q1$) of a dataset (new updated dataset here). Any data points that lie beyond the upper or lower bounds defined by

$$Q3 + 1.5 \times IQR \text{ or}$$

$$Q1 - 1.5 \times IQR, \text{ respectively, are considered outliers.}$$

Here is how the novel technique using IQR was employed for selecting samples closer to the real malicious data:

1. Large malicious samples were generated by six different types of constructed theft attacks, each with different characteristics.
2. Calculate the IQR for the real attack data, considering relevant features or characteristics of the attacks.
3. Calculate the IQR for the simulated attack data, using the same features as above, and
4. Finally only those simulated attacks were selected that lie within the range defined by $Q1 - 1.5 \times IQR$ and $Q3 + 1.5 \times IQR$ of the real attack data.

By following this process, the authors were able to identify simulated attacks that are closer to malicious samples present already in the dataset:

Normal generation of synthetic data does not match real-world theft instances and has issue of over-fitting. Theft data points are outliers since they do not fall inside the median of a normal distribution.

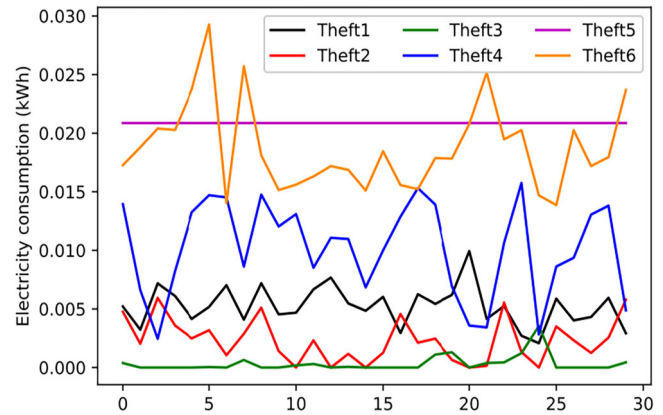


FIGURE 4 Electricity theft pattern of consumer involved in theft.

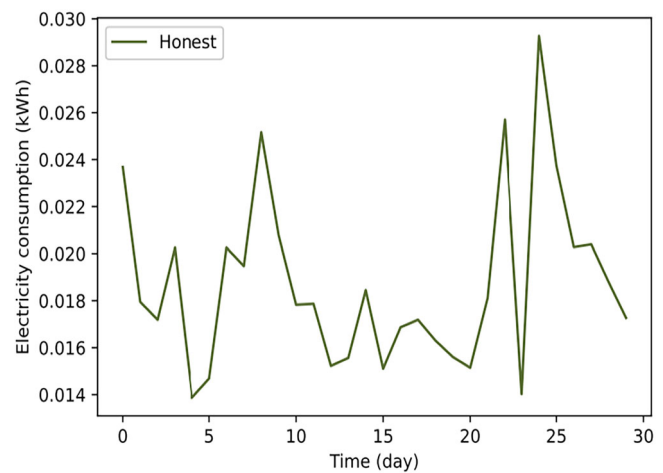


FIGURE 5 Electricity consumption pattern of honest consumer.

The application of theft attacks on benign users' consumption data was employed in order to achieve a suitable equilibrium between instances of theft and honest data points. The SGCC dataset consists of a total of 42,372 records, with 3615 instances representing abnormal consumer data and 38,757 instances representing normal consumer data. The dataset exhibits a ratio of 1:9 between normal and abnormal consumers. The dataset contains a substantial number of data points, posing challenges in utilizing all of them for analysis due to the issue of increased computational complexity. As an illustration, a total of 9999 records out of the 42,372 available were selected for the purpose of analysis. In this manner, a subset of 900 real theft records, ranging from 2714 to 3615, is chosen for analysis, while the remaining real theft records, ranging from 0 to 2713, are left out. Furthermore, the remaining deficient abnormal records, totaling 4099 in number, have been artificially generated. The attack types 1 to 6 are put into the benign consumers' data ranges from 901–1583, 1584–2266, 2267–2949, 2950–3632, 3633–4315, and 4316–4999, respectively.

The dataset comprises energy consumption data for a total of 1036 days, and the attacks are executed on the entirety of this dataset. However, Figures 4 and 5 provided below serve as

TABLE 4 Comparison of XGBoost-based detector and with SVM and LightGB model on various theft attacks for evaluation.

Attack scenario	Models	PR (%)	FPR (%)	Recall (%)	AUC (%)
Type 1	SVM	89	87	8.4	88
	LightGB	83	75	7.1	78
	Our model	94	93	7.2	92
Type 2	SVM	89	90	8.01	88
	LightGBM	80	70	17.1	73
	Our model	92	91	5.5	90
Type 3	SVM	84	87	5.4	88
	LightGBM	88	85	6.5	85
	Our model	97	94	9.33	96
Type 4	SVM	87	86	4.64	84
	LightGBM	87	85	8.0	83
	Our model	91	90	5.6	90
Type 5	SVM	89	88	5.9	82
	LightGBM	88	87	10.7	83
	Our model	95	83	6.83	87
Type 6	SVM	88	90	6.6	88
	LightGBM	89	88	6.1	86
	Our model	95	93	7.1	91
Combined	SVM	83	84	7.2	81
	LightGBM	88	86	5.7	90
	Our model	97	95	5.4	93

FPR, false-positive rate; PR, positive rate; XGBoost, extreme gradient boosting.

an example, illustrating only 30 days of synthetic attacks and normal patterns. PCA is a widely used technique for feature extraction in diverse domains, including electricity consumption datasets [24]. It aims to transform the original high-dimensional feature space into a lower-dimensional representation while preserving essential information. See Table 4 in Table 4 for results.

3.4 | Application of PCA for dimensionality reduction

In the context of electricity consumption datasets, the data is organized into a matrix format, where each row represents a sample (e.g. a day or an hour) and each column represents a feature (e.g. different electricity consumption attributes). Standardization is performed on the dataset by subtracting the mean and dividing by the standard deviation of each feature and is crucial to ensure that features are on a similar scale, as PCA is sensitive to the variances of the features [24, 25]. The covariance matrix is computed to analyze relationships between features. The eigenvectors and eigenvalues are computed from the calculated covariance by performing an eigen-decomposition on the matrix. The primary components are the new orthogonal axes in the feature space, and they are represented by the

TABLE 5 Evaluation metrics of XGBoost theft detector with and without dimensionality reduction.

Metrics	Without PCA	With PCA
Accuracy	90%	95%
Precision	85%	90%
Recall	95%	98%
F1 score	90%	94%
AUC-ROC	0.95	0.98

PCA, principal component analysis; XGBoost, extreme gradient boosting.

eigenvectors. The eigenvalues reveal how much variation is explained by each principal components PC [12].

Choosing Principal Components: Select the top k eigenvectors by sorting the eigenvalues from largest to smallest, where k is the number of major components. Most of the data's variability is captured by the top k eigenvectors, which are also the principal components [24]. The standardized data is projected onto these principal components to obtain a transformed feature space. These transformed features serve as the extracted features from the original dataset, capturing the most significant variability [24, 25]. Applying PCA to electricity consumption datasets enables dimensionality reduction while retaining important information for tasks such as visualization, anomaly detection, or theft detection [12].

As we can see in Table 5, the XGBoost model that was trained on the PCA-reduced dataset had a higher accuracy, precision, recall, F1 score, and AUC-ROC than the model that was trained on the original SGCC dataset. This suggests that PCA can be a helpful way to improve the performance of XGBoost models for ETD.

3.5 | Feature engineering

Features are extracted to add additional parameters from the existing to capture relevant patterns or relationships. Relevant features can enhance the efficiency and reliability of electricity theft detecting model [26, 27]. Various features collected directly and extracted using statistical functions include Consumer specific unique ID, electricity usage time and date (Timestamp), electricity consumption (kWh), active power, reactive power, average voltage, global intensity, power factor, max_load, min_load, average_load, load dispersion, peak demand, total load profile, seasonal variation, time-of-use, historical consumption, socio-demographic data, billing information, geographic information, time of day usage, weekday/weekend, and state holiday. In addition to the electricity consumption and electric power parameters, the utility has other features available, like a profile of the customer, tariff information, meter location, previous theft information, the consumer's maximum and minimum consumption during the last year, utility information, sub-divisional information, curtailment schedule (if any due to demand-supply gap). Apart from that, data collection is done using GIS location, load agreement,

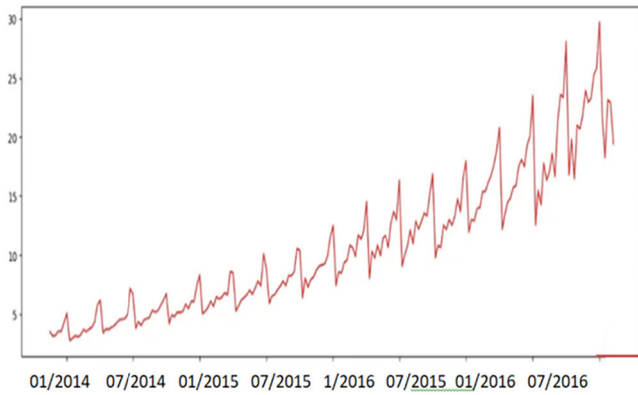


FIGURE 6 Trend in electricity usage of a consumer.

neighbourhood consumption details. Further, the weather database available (max. temp., min. temp., precipitation etc.), and the technical details (meter type, meter location etc.) are also included in the processed dataset. The extracted features also include the categorical variable to store values for time of day like (M: morning, A: afternoon, E: evening, N: night). Binary variable for weekday/weekend (0 for weekday, 1 for weekend), and binary variable for state holidays (Holiday as 1 and rest as 0). Each consumer's electricity consumption is analyzed over a period of time. The focus is on understanding the consumption patterns, identifying anomalies or deviations specific to each consumer, and detecting any unusual behaviour or theft within their consumption data. This approach allows analysis for finding the unique characteristics and consumption patterns of each consumer [26–28]. Data aggregation is performed on data consumption over different periods (daily, weekly, monthly). It involves combining the individual consumption readings within each period and calculating statistical measures or creating lag variables to capture temporal patterns [26, 28]. The following features are extracted using statistical functions:

$$\text{data_columns} = \{ \text{'smID'}, \text{'energyConsumption/hh'}, \text{'Total KWhr'} \} \quad (5)$$

Daily Aggregation: To aggregate data daily, all the consumption readings are collected to produce statistical metrics, that is, mean, variance, minimum, maximum, or sum of electricity consumption during a day for a single consumer. These parameters reveal daily average, spikes of low and high usage [26–30]. Figures 6 and 7 depict the trend of a consumer over longer and shorter periods, respectively.

Weekly Aggregation: For weekly aggregation, weekly data is combined, the statistical functions reveal weekly average, weekly consumption patterns, high and low usage days in a week.

Monthly Aggregation: The monthly aggregation groups the consumption data by month. Monthly aggregation helps in uncovering the long-term consumption trends such as seasonal fluctuations in this study.

Lag Variables: Past consumption builds lag variables and captures temporal patterns. Lag variables represent difference

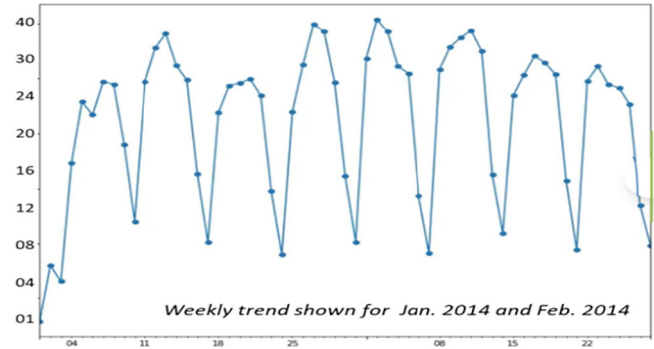


FIGURE 7 Weekly trend in electricity usage of a consumer.

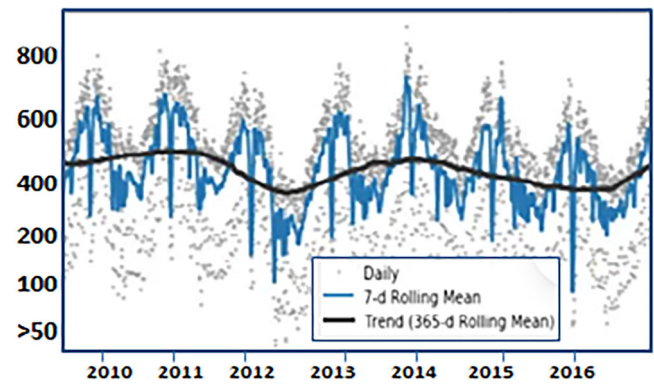


FIGURE 8 Daily, weekly, and 365-day rolling trends.

in consumption of the preceding day, preceding week, or preceding month on comparison. The lag variables assist us to identify data increase in usage over time, dependencies, and consumption patterns [26].

This study utilizes the Resampling which is a statistical technique that involves the consolidation of data within a specified timeframe. The performance of this function is similar to that of the ‘*Groupby*’ function in SQL. In other words, the data is initially divided into time bins, and subsequent computations are carried out on each bin. Resampling is done on an hourly, daily, monthly, and yearly basis to provide the relevant statistics such as minimum, maximum, and mean values in consumption [29, 30]. To compute hourly mean values for electricity consumption

```
data_columns = ['smID', 'energyConsumption/hh', 'Total
KWhr']
data_hourly_mean = data[data_columns].resample('H').
mean()
# H stands for hourly data_hourly_mean.
```

Likewise, weekly and monthly mean is calculated by using weekly (‘W’) and Monthly (‘M’) mean.

A Rolling window technique for weekly trends: The distinction between the rolling and hourly/weekly/ monthly lies in the overlapping nature of the bins [26–30]. The bins for weekly rolling resampling as shown in Figure 8 are organized as follows: 1 January to 7 January, 8 January to 14 January, 15 January to 21

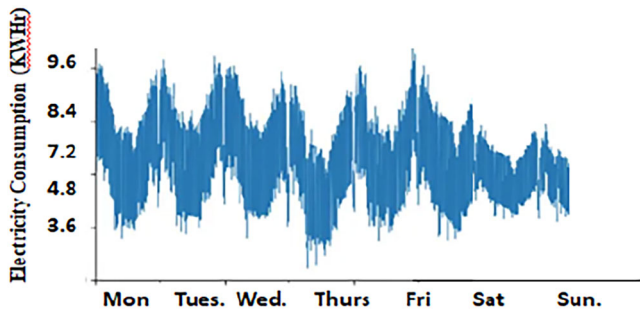


FIGURE 9 Consumers consumption pattern peak in the evenings and low during the day Also, Lower spikes during weekends.

January, and so on. The bins are organized on a weekly rolling basis, with each bin representing 7 days. For example, the first bin spans from 1 January to 7 January, the second bin spans from 2 January to 8 January, the third bin spans from 3 January to 9 January, and so on. In order to calculate a 7-day rolling mean, we follow the mathematical procedure as below [29, 30]:

$$\begin{aligned} data_7d_rol &= data[data_columns].rolling(window = 7, \\ center &= True).mean() \end{aligned} \quad (6)$$

In the above-mentioned code, the parameter `centre = True` indicates that when calculating the rolling mean for a given time bin, such as from 1 January to 8 January, the resulting value will be positioned adjacent to the middle of the bin, specifically on 4 January.

Visualizing trend in data using rolling means: Trend is the smooth long-term tendency of a time series. It might change direction (increase or decrease) as time progresses [26–30]

Seasonal trends: One effective method for visualizing the trends is through the utilization of rolling means at various time scales as shown in Figure 8. Upon analyzing Figure 8 for 365-day rolling mean time series, it becomes evident that the general annual trend in electricity consumption of a genuine consumer exhibits a considerable level of consistency [27] as pointed in Figure 9. For training a machine learning model trends need to be removed as trends can obscure the true underlying patterns in the data and can lead to spurious correlations and incorrect conclusions in statistical analyses [29, 30].

By removing the trend, the data is transformed into a stationary series, making it more amenable to the problem of theft detection on historical data. Removal of trends improves efficiency and proves advantageous, especially when the trend is prominently observable [27]. In this study, trend is eliminated using a technique called differencing. Differencing involves the creation of a data value in which the value at a given time (t) is calculated by subtracting the actual recorded reading at that time (t) from the actual recorded reading at the preceding time ($t-1$) [29, 30].

Differencing transforms non-stationary data into a stationary data. This facilitates the precise assessment of the seasonal fluctuations or random fluctuations observed in the electricity consumption time series data [27, 29, 30]. Now, the values at

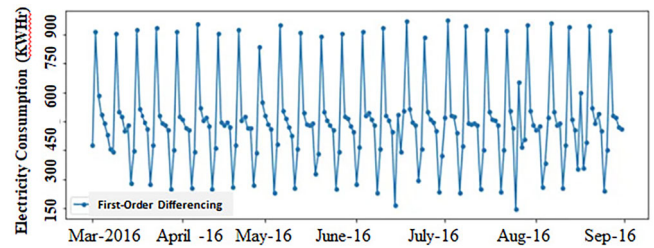


FIGURE 10 First-order differencing of consumption.

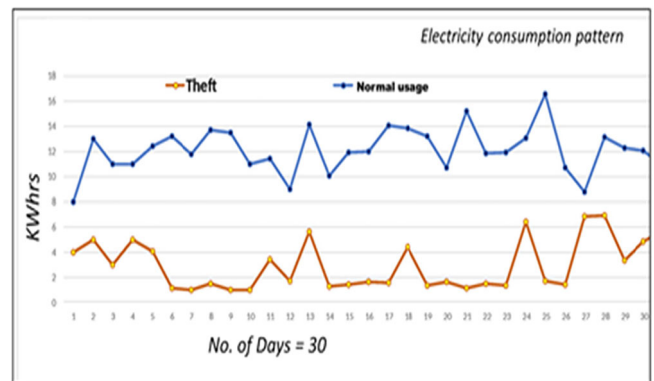


FIGURE 11 Comparison in usage trend of consumers—normal vs. fraud consumer.

this differenced column are a subtraction of two consecutive values recorded by the smart meter. In general, the information conveyed by differenced readings is not about the specific value at a given point in time, but rather the magnitude of its deviation from the previous point in time [29, 30]. Figure 10 depicts that there exists a notable peak in correlation at the seventh-day lag.

The graph is a plot of differenced values; the preponderance of the values will be distributed along both sides of the x -axis (where $y = 0$). This is due to the likelihood that most of consumption values will either be higher or lower than the previous day, and fewer instances of values where difference is zero between two consecutive days. The experiments show that the dataset has performed well on first-order differencing as demonstrated. A research study was undertaken to ascertain the existence of any weekly patterns within the dataset. Based on the depicted plot in Figure 8, the observed practice displays a prominent peak during the evening and night hours while diminishing during the daytime. Moreover, it is evident that the consumption on weekdays surpasses that on weekends, as indicated by the lower spikes observed on Saturdays and Sundays. In this study, we analyze a dataset that encompasses two months, specifically January 2014 and February 2014. The presence of distinct weekly variations is readily apparent in the observed data. The analysis of power consumption patterns reveals a notable disparity between weekends and weekdays, with the former exhibiting a lower level of energy usage and the latter characterized by significantly higher consumption rates. Monthly aggregation helps in uncovering the long-term consumption trends such as seasonal fluctuations. Figure 11 shows

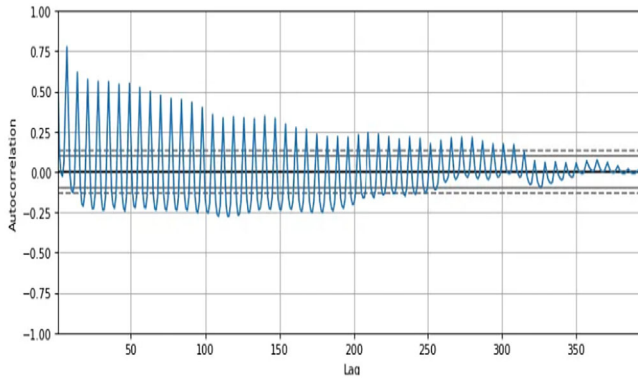


FIGURE 12 Shows the trend vanishing after about 300 days.

a comparison trend of normal and fraudulent consumer over a period of 30 days.

Autocorrelation: Autocorrelation can find seasonal trends in time series data. The autocorrelation function (ACF) is highly useful in analyzing the historical electricity to reveal seasonal patterns [29, 30]. High autocorrelation values at various lags suggest a strong link between past and future values on the daily, weekly, seasonal, or monthly consumption patterns. Aberrant usage, spot anomalies, or deviations indicate electricity [29, 30]. Autocorrelation analysis can also help in forecasting which is beyond this study. It has been seen that there is a notable peak in correlation at the seventh-day lag. Subsequently, a similar peak is observed on the fourteenth day, followed by subsequent occurrences. The observed phenomenon exhibits a repeating pattern over 7 days, indicating a weekly time series. The observed pattern exhibits a gradual decline in effectiveness over approximately three months or approximately 300 days. As the time increases, the degree of correlation between them diminishes. Figure 12 reveals that the consumption series is genuinely auto-correlated with a lag of 1 week for a specific normal consumer.

4 | MODEL COMPARISON AND SELECTION

In this section, we will provide a concise description of the models that have been implemented to conduct experiments on the SGCC dataset for the detection of electricity theft. The classification machine learning models were evaluated using a processed dataset of both benign and malicious samples. The processed dataset was not used to train the machine learning model but was used to evaluate its performance. Most of the machines learning models were able to detect electricity theft with good accuracy.

Autoregressive model: The autoregressive (AR) model is a statistical model commonly used in time series analysis like EC data [29, 30]. This linear regression model helps to predict the value of electricity consumption based on its previous values (previous half-hour consumption). The approach is used in the modelling of univariate time series [29]. A distinct unit is

employed for each unique day type, including weekdays, Saturdays, and Sundays, as well as for each hour for each consumer.

$$s_t^{b,d} = c + \sum_{i=1}^q \phi_i^{b,d} s_{t-i}^{b,d} \quad (7)$$

In this study, we consider a constant denoted as c , along with model parameters $\phi_{b,t}$.

In order to maintain simplicity and avoid a trial-and-error parameterization process, the parameters have been set to ϕ instead of making adjustments to them. In accordance with the established methodology outlined in the other research papers like [29, 30], the work-day schedule is employed as a means of categorizing the specific type of day for which load prediction is to be conducted.

In light of the aforementioned considerations, it has been deemed necessary to make certain adjustments to the prediction methodology employed for each hour. This adjustment involves the utilization of the AR model, with the specific parameters being contingent upon the type of day under consideration. Furthermore, the value of q , representing the order of the AR model, is set to 3 arbitrarily without any justification here. Therefore, it is important to acknowledge that the computation performed considered the three most recent values of the same day type. For instance, if the reference day is a Tuesday, the calculation included the values from any previous like Tuesday, Monday, Friday, or Thursday and not with the Saturday or Sunday (weekends).

Neural network: The neural network (NN) model, a type of non-linear circuit that functions as a perceptron, and which is considered a simple information processor, is also tested on the dataset. The structure of the NN is capable of adapting based on the information that is received from either external or internal sources during the learning phase. The output of the model can be described as a function that is either linear or non-linear in relation to the inputs. As a result, they have gained significant popularity in the field of predicting non-linear data, as evident by its use by the authors in [26–30] for the detection of electricity theft. After conducting the tests, the favourable outcomes were also achieved through the utilization of a NN, which incorporates temperature-related variables, the preceding hour's value (regardless of the day type), and the value of the same hour on the previous day of the same type. Furthermore, it was determined that the NN architecture required the inclusion of two concealed layers, with the first layer consisting of 35 perceptrons and the second layer consisting of 25 perceptrons.

Bayesian Networks: Bayesian Networks (BN), a probabilistic multivariate analysis framework that expands Bayes' theorem, was also employed for comparison. They use an acyclic-directed graph and a probability distribution function to represent the set of probabilistic relationships among the variables modelling the specific problem [29, 30]. The probability function shows on each node the strength of these relationships or graph edges [29, 30]. BNs have mostly been studied with discrete variables, linear Gaussian models, or combinations of both because continuous variables are difficult to represent by an estimated magnitude

TABLE 6 Proposed XGBoost-based model before and after the application of balancing techniques.

Parameter	User	Before balancing	After balancing
Precision	Genuine	0.97	0.96
	Fraud	0.56	0.94
Recall	Genuine	0.94	0.95
	Fraud	0.52	0.92
F1-Score	Genuine	0.94	0.95
	Fraud	0.42	0.93
Overall accuracy		0.90	0.94

XGBoost, extreme gradient boosting.

and range of uncertainty [29, 30]. We solved this problem by using Agglomerative Hierarchical clustering [29, 30], to cluster the load numbers for each hour and then determine the average load for each cluster. Thus, the BN classifies the load into one of those groups and predicts the average load of that class (plus the error, the difference between the actual load value and the average load of the assigned cluster) giving an anomaly in usage. Like the AR model, we created three BNs for each day. We added weather variables, day type, prior hour load value, and previous same-type day load value. For each hour, we re-trained the BN and predicted the next hour. Finally, we used the PC Algorithm [29, 30], the Expectation-Maximisation algorithm [29, 30] for parametrical learning, and the Lauritzen and Spiegelhalter method for conclusion inference over junction trees [29, 30] to achieve Bayesian inference (actual prediction) and anomaly detection in usage.

We use other alternative boosting algorithms such as LightGBM [26] and CatBoost [27] and report the corresponding results concisely in Table 4.

5 | MODEL SELECTION

To determine how well the proposed model worked, a performance comparison was performed between the XGBoost-based detector and a number of other machine learning algorithms, including k-NN, LightGB, SVM, CatBoost, Random Forest, Bayesian networks, decision tree (DT), and logistic regression. We compare the proposed model on both the imbalanced raw dataset and the balanced dataset. Table 6 shows a summary of the performance metrics for these two instances. In each case, we set the training ratio to 80%. Before the fake theft data was added to the dataset, the number of theft instances of users was fewer compared to the number of genuine users. It was because of class imbalance, the model was not able to put the fraud users into the right category because of over-fitting, as shown by the accuracy, recall, and F1 scores in the table. When fake data were added to the dataset, the model was better able to identify fraudulent users, as seen in Table 6. Experiments conducted for selection of model: In the first experiment, we trained our model on both benign and malicious samples. We randomly selected 50% of the

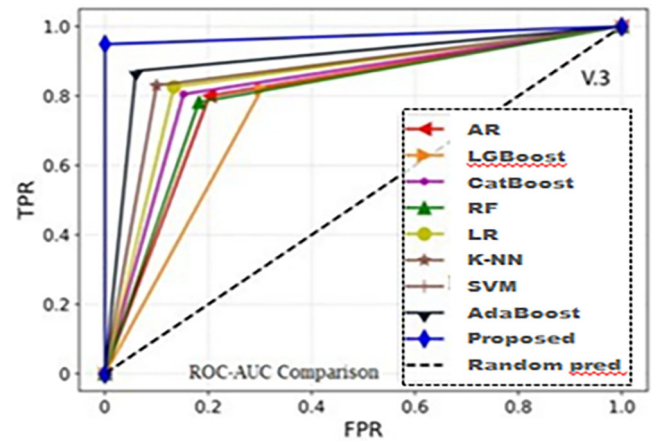


FIGURE 13 Comparison true positives vs false positives of XGBoost-based detector with benchmark ETD models. ETD, electricity theft detection; XGBoost, extreme gradient boosting.

entire 9998 samples to generate malicious samples. 70% of the samples in the dataset were picked at random as the training set, while the remaining 30% comprised the test set. We examined our method's performance in recognizing the six different attack types of electricity theft individually and it was also compared with SVM and LightGB which are next best results after our proposed XGBoost-based model. The result is seen in Table 4. This method was repeated on 35 users. In the second experiment, we trained our model using all six attack types and compared it to two best commonly used AI-based methods: SVM and LGBost. Lastly, five pairs of parameters as shown are examined (randomly generated variable for six attack types) to evaluate the influence of different α and α_t values under different attack types (Attack Type 1, Attack Type 2, and Attack Type 4): (0.3, 0.8), (0.4, 0.8), (0.5, 0.8), (0.6, 0.8), and (0.7, 0.8) [26–30]. In Figure 4 above the XGBoost-based detection model distinguishes different attack types with good accuracy for each pair of t values. Table 4 displays the findings of the experiment.

A comparison of true positives and false positives of an XGBoost-based detector with benchmark ETD models is shown in Figure 13. The precision-recall curves are shown in Figures 14 and 16. The ROC-AUC (between true positives and false positives) curve of the proposed detector is shown in Figures 15 and 17.

5.1 | XGBoost model overview

In the context of machine learning methods, the objective function is the sum of the loss function (L) and the regularization term (Ω) over the parameters (θ) [6, 22].

$$\text{Obj}(\theta) = L(\theta) + \Omega(\theta) \quad (8)$$

The XGBoost's objective function (derived from Equation (8)) combines the sum of a specific loss function (L) evaluated over all n predictions (or samples) and the sum of a

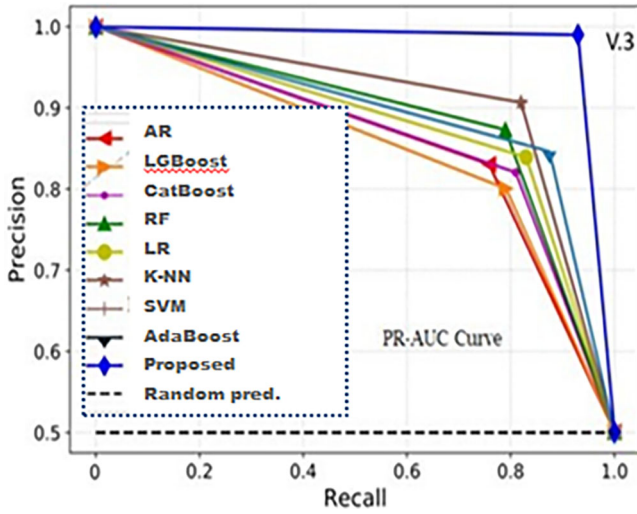


FIGURE 14 Comparison of precision vs recall values of XGBoost-based detector with benchmark ETD models. ETD, electricity theft detection; XGBoost, extreme gradient boosting.

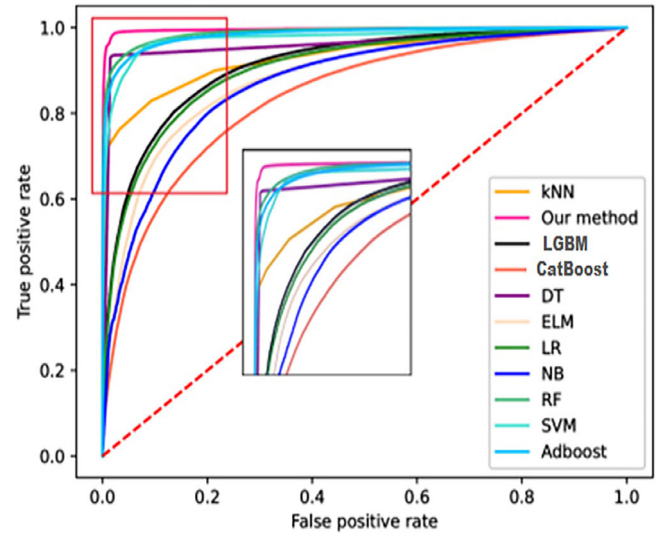


FIGURE 17 Comparison of true positives vs false positives of XGBoost-based detector with benchmark ETD models. ETD, electricity theft detection; XGBoost, extreme gradient boosting.

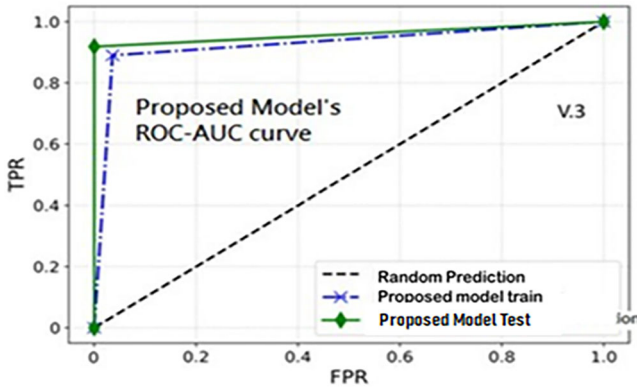


FIGURE 15 Comparison true positives vs false positives (ROC-AUC) of XGBoost-based detector with benchmark ETD models. ETD, electricity theft detection; XGBoost, extreme gradient boosting.

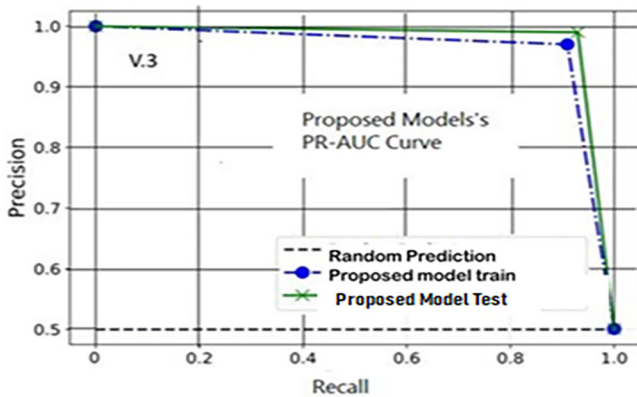


FIGURE 16 Comparison of precision vs recall (PR-AUC) of XGBoost-based detector with benchmark ETD models. ETD, electricity theft detection; XGBoost, extreme gradient boosting.

regularization term (Ω) for all predictors (K DTs) as follows:

$$Obj(\theta) = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (9)$$

In Equation (9), f_k represents the k th DT function, y_i denotes the actual label of the i th sample, and \hat{y}_i represents the predicted label of the i th sample. The DT structure and objective function are further elucidated in the comprehensive XGBoost hyperparameter tuning guide [6, 22]. The primary objective of the classifier algorithm is to minimize the value of the objective function as expressed in Equation (9) [6]. The loss function in Equation (9) may encompass various options, such as the log-loss function, squared loss function, or other alternatives. The prediction error of the machine learning model is governed by the control mechanism, whereas the complexity of the model is regulated by the regularization term Ω , which adjusts factors such as the size of the tree structure and the depth of the trees [6, 22].

5.2 | Working of XGBoost

Given a classification task, a data set D can be represented as

$$D = \{(x_i, y_i)\} (|D| = n, x \in R^m, y_i \in R) \quad (10)$$

where D represents a given dataset, x_i denotes a vector consisting of n samples and m features, and y_i represents the corresponding label [6]. In this article, the symbol x_i represents the measurement of the meter reading during a specific day period, which consists of n samples. The variable y_i is defined as a binary value, where $y_i = 0$ signifies that the user does

not exhibit any abnormal power usage behaviour, and $y_i = 1$ indicates noticed abnormal or malicious activity [6].

The representation (as seen in Equation (11)) of a tree ensemble model utilizing K additive functions f_k is as follows:

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), f_k = F \quad (11)$$

Let F denote the set of functions that encompasses all classification trees. Instead of collecting the weights within the tree model, XGBoost employs the process of learning functions [6]. The objective function of XGBoost is formulated as follows:

$$Obj = \sum_i l(y_i, \hat{y}_i) + \sum_k \Omega(f_k) \quad (12)$$

with $\Omega(f) = \alpha T + \frac{1}{2} w^2$

where the loss function denoted by l quantifies the degree of alignment between the model's predictions (\hat{y}_i) and the corresponding target values y_i in the training dataset. The regularization term quantifies the level of complexity exhibited by the model. XGBoost incorporates L1 and L2 regularization terms into the gradient-boosting DT framework. The variable T denotes the score of leaf nodes, while the variable w represents the scores assigned to the said leaf nodes. The inclusion of a regularization term provides a valuable advantage in mitigating the issue of overfitting [6, 31].

The training of the model is conducted in an additive manner [31]. Let $\hat{y}_i^{(t)}$ be the prediction term of the i th instance at the t th iteration. The optimization of the objective of the i th instance at the t th iteration can be achieved [6].

$$\mathcal{L}^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (13)$$

$$\mathcal{L}^{(t)} = \left[\sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + g_i f_i(x_i) + \frac{1}{2} b_i f_i^2(x_i)) \right] + \Omega(f_i) \quad (14)$$

where g_i and b_i are the first- and second-order gradient stochastics on the loss function, respectively.

$$g_i = \partial \hat{y}_i^{(t-1)} l(y_i, \hat{y}_i^{(t-1)}) \quad (15)$$

$$b_i = \partial^2 \hat{y}_i^{(t-1)} l(y_i, \hat{y}_i^{(t-1)}) \quad (16)$$

I_j is defined as $\{i \mid q(x_i) = j\}$, representing the collection of instances that are assigned to leaf j . After eliminating all the constant terms and extending the equation, the specific objective at step t is transformed [6].

$$\mathcal{L}^{(t)} = \sum_{i=1}^n \left[g_i f_i(x_i) + \frac{1}{2} b_i f_i^2(x_i) \right] + \Omega(f_i) \quad (17)$$

TABLE 7 Hyperparameter tuning of the proposed XGBoost model.

Hyper parameter	Value	Description
Loss function	Binary:logistic	Binary classification
Booster	Gbtree	
learning_rate	0.05	
n_estimators	100	
Maximum depth of trees	8	
reg_lambda	2	L2 regularization term on weights

XGBoost, extreme gradient boosting.

$$\mathcal{L}^{(t)} = \sum_{i=1}^n \left[g_i f_i(x_i) + \frac{1}{2} b_i f_i^2(x_i) \right] + \gamma T + \frac{1}{2} \pi \sum_{j=1}^T w_j^2 \quad (18)$$

$$= \sum_{j=1}^T \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} b_i + \tau w_j^2 \right) \right] + \gamma T \quad (19)$$

The optimization objective for the new tree is represented by Equation (19), which is utilized by XGBoost to facilitate the incorporation of custom loss functions. One notable benefit of XGBoost is its incorporation of regularizations into the loss function [6, 31]. This inclusion results in the creation of simpler trees and serves as a preventive measure against overfitting. XGBoost generates additional trees in order to rectify the remaining errors in the predictions made by the current sequence of trees. The observed outcome is that the model demonstrates a rapid ability to fit into the training dataset, subsequently leading to overfitting [6, 26–31]. Achieving a balanced fitting performance is typically accomplished by adjusting the hyperparameters of the machine learning model during both the training and testing phases [6]. Furthermore, the XGBoost classifier is equipped with a module called ‘feature_importance’ that allows for a comprehensive understanding of the classifier model. This module provides a feature score, also known as an f -score, for each individual feature. The hyperparameter tuning is discussed in the next section.

5.3 | Hyperparameters tuning

Table 7 provides a concise explanation of the hyperparameters employed in the XGBoost-based ETD model. It also enumerates the specific values associated with these hyperparameters.

Parallel Threads: In XGBoost, we set the training thread count ($nthread$) to 2. This parameter determines how many CPU cores the model uses for training. To maximize training speed, we set it to the number of available cores.

Iteration Count: During training, XGBoost builds boosting rounds or trees based on iterations ($n_estimators$ or num_boost_round). We use cross-validation to find the best iteration number. The model was trained of training set with different iterations and evaluation of performance was done on the validation set. This was done in order to find an ideal value

TABLE 8 Confusion matrix.

Confusion matrix	Actually positive	Actually negative
Predicted positive	True positives (TPs)	False positives (FPs)
Predicted negative	False negatives (FNs)	True negatives (TNs)

where the performance stabilizes and degrades thereafter. Our optimal performance was achieved with 5-fold cross-validation.

Learning Rate: The learning rate in XGBoost (eta or learning rate) was set at 0.05. This learning rate determines the step size for each iteration when minimizing the loss function. Smaller learning rates lead to more accurate models but may require more iteration.

To discover the best learning rate, the values that were tried are: 0.01, 0.03, 0.05, 0.1, and 0.3 to evaluate the model's performance.

Regularization Parameters: XGBoost has two regularization parameters, lambda and alpha. Regularization helps prevent overfitting and promotes better generalization of the model [26–30]. To find the optimal performance on validation data, we performed grid search or random search to explore different lambda and alpha values. The value was set to 2. Furthermore, the authors in reference [26–30] assert that a lower FPR coupled with a reasonable DR serves as a reliable indicator of an effective intrusion or theft detection system. It should be noted that the hypermeter values have been established based on guidelines outlined in references [26–30] through the utilization of a grid search approach.

Resource utilization: All the experiments in this study were performed on a PC with an i7-8550U CPU and 16-GB RAM. The programming work was done using Jupyter Notebook of Python.

6 | MODEL EVALUATION

The objective of this study is to detect the anomaly or deviation in electricity consumption based on the usual pattern of electricity usage that was observed in the given time series data instances.

A confusion matrix is used to check the performance of a model [12]. For the purpose of training, we will utilize a dataset spanning a period of 02 years, specifically from 2014 to 2016. Subsequently, after the train-test split on a dataset, to evaluate the proposed model's performance, the dataset is split into three reduced sub-sets: (1) A training dataset, (2) a validation dataset to tune the hyper-parameters, and (3) a testing dataset for generalizing our model. Nested Cross Validation (NCV) is preferred for its realistic view on model generalization [32].

A confusion matrix as seen in Table 8 can be used to evaluate the performance of a model. The root mean square error (RMSE) can be employed to evaluate XGBoost-based models for ETD. The RMSE is calculated by comparing model predictions to actual electricity use. Model performance

improves with decreasing RMSE values. To identify electricity theft, an XGBoost model is trained on electricity.

See Table 9 in Table 4

use data. The model can then predict a group's electricity use as shown in Table 9. Comparing the expected and actual values yields the RMSE. RMSE can be calculated mathematically

$$sMAPE = \frac{100}{n} + \sum_{t=1}^n \frac{A_t - P_t}{A_t + P_t} \quad (20)$$

where A_t is actual consumption and P_t is the predicted value at time t .

7 | DISCUSSIONS AND COMPARISON

ETD is an extremely challenging task, as the detection does not solely depend on determining the anomaly but on all the other data forming any relationship with electricity consumption. AMIs are susceptible to cyber-attacks aimed at stealing electricity. The researchers in [11, 26–30] suggest the use of CNN-LSTM Based approach for the detection of electricity theft in contrast to the use of single-machine learning algorithms for the detection of physical theft attacks on AMIs. This study utilizes the XGBoost ensemble method for the detection of physical electricity theft attacks. The used by authors in [6] and use the electricity consumption data extracted from smart meters and a few other features from auxiliary databases to detect theft. Our approach uses features extracted from the GIS location, weather database, and features extracted from auxiliary databases. For the predictions, our model had 98% precision. Figures 13–17 show the AUC scores of our model as compared to the other popular ETD models. The proposed XGBoost-based detector was employed on real instances of theft. Our approach falls short of the high granularity used by the authors in [8] and [33] to detect intermittent fraud. In this study, it is proven that these extra features are important for detecting theft in Indian cities, since studying how consumers use electricity-run gadgets is not enough to find a wide range of NTL. However, the high granularity of EC data will lead to privacy intrusions for consumers. So, in the future, an intermediate approach is adopted by the proposed model to have more privacy preservation and lesser detection time than the current proposed version of the XGBoost-based model.

8 | CONCLUSION

As per the research studies, AMI of the smart grid helps to detect electricity theft efficiently and accurately [11, 26–30]. This study proposes an XGBoost-based ETD system where, in addition to electricity consumption data, other aspects related to usage are used. These include seasonality, location, power curtailments to cater to high demand, regional festivals, weekends, and weekdays. These additional variables improved the DRs and reduced the number of false positives in this research. By

TABLE 9 NCV of XGBoost-based detector for evaluation.

Folds	Accuracy	Recall	Precision	F1 score	Kappa	MCC
1	0.92	0.91	0.93	0.96	0.86	0.86
2	0.93	0.92	0.94	0.95	0.87	0.87
3	0.92	0.91	0.93	0.96	0.87	0.87
4	0.93	0.91	0.94	0.95	0.86	0.86
5	0.93	0.92	0.95	0.96	0.87	0.87
<i>Mean</i>	0.93	0.92	0.95	0.95	0.86	0.86
<i>Std Dev.</i>	0.00216	0.00299	0.00235	0.00205	0.00424	0.00427

NCV, nested cross validation; XGBoost, extreme gradient boosting

artificially creating six different theft attacks, we successfully mitigate dataset imbalance, ensuring a balanced representation of theft and non-theft instances. The utilization of the XGBoost algorithm for classification demonstrated outstanding performance in distinguishing between malicious and normal electricity usage, yielding high accuracy rates and a remarkably low false-positive occurrence. Our model proves effective in identifying region-specific electricity theft, utilizing various electricity consumption parameters and input features. Comparing our model to existing benchmarks like support vector machine *K*-NN, LightGBM, CatBoost, Random forest, LR, SVM, NB, DT, NN, and AdaBoost, the XGBoost-based detection model emerges as the top-performing solution. The inclusion of false attacks for dataset balancing further improves the model's performance, achieving impressive F1-score, precision, and recall rates of 97%, 98%, and 98%, respectively. The results of our research demonstrate the efficacy and reliability of the proposed XGBoost-based model in detecting electricity theft. With a DR of 96% and a minimal FPR of 3%, our approach holds great promise for utility providers in combating electricity theft and reducing financial losses. The successful application of our model showcases its potential to make a significant impact in the energy industry, safeguarding revenue streams and enhancing the security and efficiency of electricity distribution systems. Carefully identifying and analysing the input features is essential to fully understanding the ETD problem and to getting better results.

9 | FUTURE RESEARCH

This research can be expanded to address or improve the following capabilities in future research:

Privacy preservation: Though in this research, consumer's personal data is not revealed anywhere, for more security, a few privacy-preserving techniques can be incorporated to protect consumer data like meter ID, location etc. In this study, cyber-theft attacks are not taken care of. However, cyber security hardware and software, which are built into the power infrastructure and techniques, can be added to combat sophisticated cyber-theft attacks.

Advanced Feature Engineering In the future, research may include socio-economic indicators and analyse indus-

trial, commercial, and residential sectors separately due to differences in potential consumption patterns. A dynamic model can be adapted for changing electricity consumption patterns on adding new electrical gadgets in a house. This research can integrate energy consumption forecasting models with theft detection systems in the future.

AUTHOR CONTRIBUTIONS

Asif Iqbal Kawoosa: Conceptualization, Data curation, Formal analysis, Methodology, Visualization, Writing—original draft. Deepak Prashar: Data curation, Formal analysis, Investigation, Writing-review & editing. M. Faheem: Project administration, Software, Data curation, Data validation, Editing. Nishant Jha: Investigation, Software, Validation. Arfat Ahmad Khan: Funding acquisition, Resources, Supervision.

ACKNOWLEDGEMENTS

The authors are highly grateful to their affiliated universities and institutes for providing research funding and facilities.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The dataset employed in this research is available online at <https://github.com/henryRDlab/ElectricityTheftDetection> (accessed on 8th February 2023).

ORCID

Muhammad Faheem  <https://orcid.org/0000-0003-4628-4486>

Nisbant Jha  <https://orcid.org/0000-0003-2758-1215>

REFERENCES

1. Mashima, D., Cárdenas, A.A.: Evaluating electricity theft detectors in smart grid networks. In: International Workshop on Recent Advances in Intrusion Detection, pp. 210–229 (2012)
2. Messinis, G.M., Hatzigiorgi, N.D.: Review of non-technical loss detection methods. *Electr. Power Syst. Res.* 158, 250–266 (2018). <https://doi.org/10.1016/j.epsr.2018.01.005>
3. Jiang, R., Lu, R., Wang, Y., Luo, J., Shen, C., Shen, X.: Energy-theft detection issues for advanced metering infrastructure in smart grid. *Tsinghua Sci. Technol.* 19(2), 105–120 (2014). <https://doi.org/10.1109/TST.2014.6787363>

4. Salinas, S.A., Li, P.: Privacy-preserving energy theft detection in microgrids: A state estimation approach. *IEEE Trans. Power Syst.* 31(2), 883–894 (2016). <https://doi.org/10.1109/TPWRS.2015.2406311>
5. Yip, S.C., Wong, K.S., Hew, W.P., Gan, M.T., Phan, R.C.W., Tan, S.W.: Detection of energy theft and defective smart meters in smart grids using linear regression. *Int. J. Electr. Power Energy Syst.* 91, 230–240 (2017). <https://doi.org/10.1016/j.ijepes.2017.04.005>
6. Yan, Z., Wen, H.: Electricity theft detection base on extreme gradient boosting in AMI. *IEEE Trans. Instrum. Meas.* 70, 1–9 (2021). <https://doi.org/10.1109/TIM.2020.3048784>
7. Henriques, H.O., et al.: Development of adapted ammeter for fraud detection in low-voltage installations. *Measurement* 56, 1–7 (2014)
8. Cárdenas, A.A., Amin, S., Schwartz, G., Dong, R., Sastry, S.: A game theory model for electricity theft detection and privacy-aware control in AMI systems. In: 2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton), pp. 1830–1837 (2012)
9. Zhou, Z., Bai, J., Dong, M., Ota, K., Zhou, S.: Game-theoretical energy management design for smart cyber-physical power systems. *Cyber-Physical Syst.* 1(1), 24–45 (2015)
10. 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.* 209, 107975 (2022)
11. Faheem, M., Umar, M., Butt, R.A., Raza, B., Ngadi, M.A., Gungor, V.C.: Software defined communication framework for smart grid to meet energy demands in smart cities. In: 2019 7th International Istanbul Smart Grids and Cities Congress and Fair (ICSG), IEEE, pp. 51–55 (2019)
12. 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.* 10(9), 3151 (2020)
13. Abdallah, A., Shen, X.: Lightweight security and privacy preserving scheme for smart grid customer-side networks. *IEEE Trans. Smart Grid* 8(3), 1064–1074 (2017). <https://doi.org/10.1109/TSG.2015.2463742>
14. Muniz, C., Figueiredo, K., Vellasco, M., Chavez, G., Pacheco, M.: Irregularity detection on low tension electric installations by neural network ensembles. In: 2009 International Joint Conference on Neural Networks, pp. 2176–2182 (2009)
15. Irfan, M., et al.: Energy theft identification using adaboost ensembler in the smart grids. *Comput. Mater. Contin.* 72(1), 2141–2158 (2022). <https://doi.org/10.32604/cmc.2022.025466>
16. Ahmad, T., Chen, H., Wang, J., Guo, Y.: Review of various modeling techniques for the detection of electricity theft in smart grid environment. *Renew. Sustain. Energy Rev.* 82, 2916–2933 (2018). <https://doi.org/10.1016/j.rser.2017.10.040>
17. Amin, S., Schwartz, G.A., Cardenas, A.A., Sastry, S.S.: Game theoretic models of electricity theft detection in smart utility networks. *IEEE Control Systems Magazine.* 35(1), 66–81 (2015)
18. Leite, J.B., Mantovani, J.R.S.: Detecting and locating non-technical losses in modern distribution networks. *IEEE Trans. Smart Grid* 9(2), 1023–1032 (2018). <https://doi.org/10.1109/TSG.2016.2574714>
19. Wang, S., Chen, H.: A novel deep learning method for the classification of power quality disturbances using deep convolutional neural network. *Appl. Energy* 235, 1126–1140 (2019). <https://doi.org/10.1016/j.apenergy.2018.09.160>
20. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794 (2016)
21. Qu, Z., Liu, H., Wang, Z., Xu, J., Zhang, P., Zeng, H.: A combined genetic optimization with AdaBoost ensemble model for anomaly detection in buildings electricity consumption. *Energy Build.* 248, 111193 (2021). <https://doi.org/10.1016/j.enbuild.2021.111193>
22. Punmiya, R., Choe, S.: ToU pricing-based dynamic electricity theft detection in smart grid using gradient boosting classifier. *Appl. Sci.* 11(1), 1–15 (2021). <https://doi.org/10.3390/app11010401>
23. Pamir, et al.: Synthetic theft attacks and long short term memory-based preprocessing for electricity theft detection using gated recurrent unit. *Energies* 15(8), 2778 (2022). <https://doi.org/10.3390/en15082778>
24. Krishna, V.B., Weaver, G.A., Sanders, W.H.: PCA-based method for detecting integrity attacks on advanced metering infrastructure. In: International Conference on Quantitative Evaluation of Systems, pp. 70–85 (2015)
25. Faheem, M., Butt, R.A.: Big datasets of optical-wireless cyber-physical systems for optimizing manufacturing services in the internet of things-enabled industry 4.0. *Data Brief.* 42, 108026 (2022)
26. Jokar, P., Arianpoo, N., Leung, V.C.M.: Electricity theft detection in AMI using customers' consumption patterns. *IEEE Trans. Smart Grid* 7(1), 216–226 (2015)
27. Sardianos, C., et al.: Reshaping consumption habits by exploiting energy-related micro-moment recommendations: A case study. In: International Conference on Smart Cities and Green ICT Systems, pp. 65–84 (2019)
28. Jokar, P., Arianpoo, N., Leung, V.C.M.: Electricity theft detection in AMI using customers' consumption patterns. *IEEE Trans. Smart Grid* 7(1), 216–226 (2016). <https://doi.org/10.1109/TSG.2015.2425222>
29. Himeur, Y., Ghanem, K., Alsalemi, A., Bensaali, F., Amira, A.: Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives. *Appl. Energy* 287, 116601 (2021). <https://doi.org/10.1016/j.apenergy.2021.116601>
30. Rashid, H., Singh, P.: Poster abstract: Energy disaggregation for identifying anomalous appliance. In: BuildSys 2017 - Proceedings of the 4th ACM International Conference System Energy-Efficient Built Environ, vol. (2), pp 2–3 (2017). <https://doi.org/10.1145/3137133.3141438>
31. Chen, X., Qiu, X., Ma, Y., Wang, L., Fang, L.: Boruta-XGBoost electricity theft detection based on features of electric energy parameters. *J. Phys. Conf. Ser.* 2290(1), 012121 (2022). <https://doi.org/10.1088/1742-6596/2290/1/012121>
32. Buzau, M.M., Tejedor-Aguilera, J., Cruz-Romero, P., Gomez-Exposito, A.: Detection of non-technical losses using smart meter data and supervised learning. *IEEE Trans. Smart Grid* 10(3), 2661–2670 (2019). <https://doi.org/10.1109/TSG.2018.2807925>
33. Schütte, S., Scherfke, S., Tröschel, M.: Mosaik: A framework for modular simulation of active components in Smart Grids. In: 2011 IEEE 1st International Workshop on Smart Grid ModelIng and Simulation, SGMS 2011, pp. 55–60 (2011). <https://doi.org/10.1109/SGMS.2011.6089027>

How to cite this article: Kawoosa, A.I., Prashar, D., Faheem, M., Jha, N., Khan, A.A.: Using machine learning ensemble method for detection of energy theft in smart meters. *IET Gener. Transm. Distrib.* 1–16 (2023). <https://doi.org/10.1049/gtd.12997>