




ORIGINAL RESEARCH

A novel two-stage method to detect non-technical losses in smart grids

Sufian A. Badawi¹ | Maen Takruri²  | Mahmood G. Al-Bashayreh¹ |
 Khoulood Salameh³ | Jumana Humam³ | Samar Assaf³ | Mohammad R. Aziz⁴  |
 Ameera Albadawi⁵ | Djamel Guessoum^{2,6} | Isam ElBadawi⁷ | Mohammad Al-Hattab⁸ 

¹Department of Computer Science, Faculty of Information Technology, Applied Science Private University, Amman, Jordan

²Center of Information, Communication and Networking Education and Innovation (ICONET), American University of Ras Al Khaimah, Ras Al Khaimah, United Arab Emirates

³Department of Computer Science and Engineering, American University of Ras Al Khaimah, Ras Al Khaimah, United Arab Emirates

⁴Department of Electrical Engineering, College of Engineering, American University of Sharjah, Sharjah, United Arab Emirates

⁵Department of Computer Science, College of Computing and Informatics, University of Sharjah, Sharjah, United Arab Emirates

⁶Ecole de Technologie Superieure, Electrical Engineering Department, Montreal, Quebec, Canada

⁷Industrial Engineering Department, College of Engineering, University of Ha'il, Ha'il, Saudi Arabia

⁸College of Engineering, Al Ain University, Al Ain, UAE

Correspondence

Sufian A. Badawi.

Email: s_badawi@asu.edu.jo

Maen Takruri.

Email: maen.takruri@aurak.ac.ae

[Correction added on 4 April 2024, after first online publication. The affiliation for Djamel Guessoum is corrected as below:

2. Center of Information, Communication and Networking Education and Innovation (ICONET), American University of Ras Al Khaimah, Ras Al Khaimah, United Arab Emirates

6. Ecole de Technologie Superieure, Electrical Engineering Department, Montreal, Quebec, Canada].

Abstract

Numerous strategies have been proposed for the detection and prevention of non-technical electricity losses due to fraudulent activities. Among these, machine learning algorithms and data-driven techniques have gained prominence over traditional methodologies due to their superior performance, leading to a trend of increasing adoption in recent years. A novel two-step process is presented for detecting fraudulent Non-technical losses (NTLs) in smart grids. The first step involves transforming the time-series data with additional extracted features derived from the publicly available State Grid Corporation of China (SGCC) dataset. The features are extracted after identifying abrupt changes in electricity consumption patterns using the sum of finite differences, the Auto-Regressive Integrated Moving Average model, and the Holt-Winters model. Following this, five distinct classification models are used to train and evaluate a fraud detection model using the SGCC dataset. The evaluation results indicate that the most effective model among the five is the Gradient Boosting Machine. This two-step approach enables the classification models to surpass previously reported high-performing methods in terms of accuracy, F1-score, and other relevant metrics for non-technical loss detection.

KEYWORDS

artificial intelligence, data analytics and machine learning, data structures, power metres, power system security, smart cities, smart power grids

1 | INTRODUCTION

Smart cities are contemporary urban settings that employ state-of-the-art technologies and data-centric approaches to foster sustainability, optimise the distribution of resources, and

improve the overall well-being of inhabitants. A pivotal component of this paradigm shift is the smart grid, which is an innovative electrical distribution system that serves as the foundation for the energy infrastructure of the municipality. By means of integrating renewable energy sources, improving

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs](https://creativecommons.org/licenses/by-nc-nd/4.0/) License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2024 The Authors. *IET Smart Cities* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

efficiency, and promoting environmental sustainability, the smart grid reorients energy management. Smart meters are of paramount importance in this ecosystem as they enable the accumulation of data in real-time, thereby providing residents with the ability to oversee and regulate their electricity usage. Furthermore, these metres provide utilities with the ability to promptly identify and resolve concerns such as energy theft, power disruptions, and other sources of energy loss, thereby ensuring a dependable and secure energy provision.

Two distinct types of electrical power losses are technical loss and non-technical loss (NTL) [1]. Arising during energy transmission from power generation facilities to end-users, technical losses can be transformed into various energy forms, including heat, within power lines and transformers [2]. A decrease in technical losses may be made possible by changing and redesigning the grid's elements [3]. A NTL involves distributed energy that may not be billed due to energy theft or fraud. In the majority of nations, the primary causes of NTLs are the manipulation of smart meters and unauthorised changes to their recorded data [4]. As such, utility companies devote substantial resources and exert significant effort to detect fraudsters and protect power infrastructures from unauthorised access.

It is possible to successfully minimise NTLs by employing a variety of techniques to generate results that reveal analysable information when detecting malicious users. It is also believed that focusing on NTLs minimisation before addressing technical losses is more useful [5]. Commonly, researchers divide NTLs into two different groups: 1) tampering with the metre's readings to inflate reported usage or bypassing the metre altogether for technical official testing, and 2) NTLs resulting from personal manipulations, which involves promoting fraudulent and dishonest behaviour among the power utility or company staff [4]. Following this, NTL detection techniques can be categorised into two groups: hardware-based methods and alternative (non-hardware) approaches. To identify fraudsters, hardware-based methods utilise intelligent equipment deployed at specific locations within the electrical systems. Alternative approaches to hardware-based methods include network-based methods, data-based methods, or a mixture of the two, which are all presented in the existing research framework. These approaches have emerged alongside the development of digital technology, enabling the collection of diverse data regarding user consumption habits.

This work studies data-based techniques using machine learning (ML) algorithms, an increasingly trending area in NTL detection research primarily due to the advantages demonstrated over conventional hardware-based methods [6]. The following supervised ML algorithms were tested: Gradient Boosting Machine (GBM), Generalised Linear Model (GLM), Deep Learning (DL), and Naive Bayes (NB). The tests were conducted in a two-step analysis of the dataset provided by the State Grid Corporation of China (SGCC). In the first step, abrupt and anomalous changes in the consumer's typical usage are detected. This is done by statistically examining fluctuations in consumption patterns relative to the evolving measurement of the midpoint of a sliding timeframe. The time series analysis

method known as Auto-regressive Integrated Moving Average (ARIMA) anticipates a consumer's upcoming consumption behaviours through an assessment of past usage patterns [7]. Building upon the results from the first step, a novel attribute collection is curated for the aforementioned algorithms. The second step involves using the aforementioned ML algorithms to determine whether the activities of the consumer under consideration are deceitful. The main contributions and goals of this work are as follows:

1. The introduction of a novel technique for transforming data from smart meter readings that integrates the sum of finite differences to model the sudden jump in electricity usage, ARIMA, and Holt-Winters approaches to extract statistical characteristics to detect NTLs. The suggested dataset transformation improves the fraud detection efficacy of the tested ML algorithms mentioned above.
2. One main advantage of the proposed solution is the ability to extract the finite differences as features in the transformed dataset and the power of the GBM algorithm in detecting NTL fraud cases.
3. The evaluation of the proposed strategy and comparing it to current methods in the literature that also utilise the SGCC dataset. The experimental findings indicate that the suggested technique outperforms the state-of-the-art methods in terms of accuracy, precision, recall, and F1 score.

The rest of the paper is organised into six further sections. Section 2 reviews recent academic publications that have predominantly employed supervised machine-learning algorithms on the SGCC dataset. The selected analysis outcomes, classification methodologies, and characteristics are also mentioned. Section 3 provides details of the technique proposed in this paper, along with the materials and methods used. Section 4 outlines the steps of dataset preparation and the definitions of the performance measures used to test the performance of each algorithm. Section 5 presents the results of the tests. Section 6 discusses the results. Finally, section 7 concludes this study.

2 | LITERATURE REVIEW

The techniques used to detect and recognise cases of NTLs by fraudulent customers on victimised consumers are divided into two categories: hardware-based and non-hardware-based techniques. The electricity supplier or utility company can identify potentially fraudulent customer conduct [8, 9] by simply setting up metres, which forms the foundation for the hardware of the detection system. There are three categories of NTL identification techniques that are alternatives to hardware-based methods: 1) approaches based on data 2) techniques utilising networks and 3) methods that may be mixtures of the previous two [6, 10, 11]. Methods that use networks collect information gathered by the networks via observer metres, smart meters, and sensors strategically located

in the grid. Various review articles and related literature [6, 12] have exhaustively investigated and documented these techniques. The existing literature classifies network-based techniques into four different groups:

1. Load Flow Approach [13, 14]: the electrical power usage by customers is monitored by setting an observer metre in a designated grid spot. The result of this analysis is then contrasted with the readings from customers' smart meters, allowing for the identification of fraudulent activities.
2. Condition Prediction Methodology [9, 15]: the grid is tracked using information from smart meters.
3. Sensing Network Methodology [16, 17]: specialised sensors are implemented at specific grid spots to monitor usage.

Data-driven approaches utilise the information gathered from smart meters for a certain observation period and employ a variety of ML algorithms for obtaining individual behavioural insights. Machine learning techniques offer several advantages, such as higher accuracy, enhanced effectiveness, decreased time consumption, and decreased labour demands [6]. Detection methods using ML are classified as being either supervised or unsupervised [18, 19]. Unsupervised methods for NTL detection utilise unannotated customer data. On the other hand, supervised techniques use data that is assigned labels, such as "Fraud" or "Non-fraud". Artificial Neural Networks (ANN), Support Vector Machines (SVM), Convolutional Neural Networks (CNN), Optimum Path Forest, Decision Trees (DT), as well as various Ensemble Learning methods such as AdaBoost, XGBoost, and Random Forest (RF), represent some of the most extensively evaluated supervised techniques in the current body of literature.

SVMs are commonly used as the primary technique for NTL detection [14, 20]. While they excel with small datasets, their efficacy decreases with large, unbalanced datasets [21]. In some investigations, the SVM accuracy varied between 86% and 98% [22, 23]. A Multilayer Perceptron is also a prominent and frequently applied technique for NTL detection [2]. It has been used in [24, 25] where characteristics based on statistics and spectral analysis of the time-series dataset were employed, yielding accuracy ratings that vary from 54.61% to 87.11%.

In their study [26], Hussain et al. presented CatBoost, an innovative method to identify NTLs based on supervised ML. The researchers used it to analyse the SGCC dataset. The SGCC dataset is an authentic dataset made publicly available by the SGCC and includes the electricity consumption data of 42,372 customers over a duration of 1035 days. Moreover, the Feature Extraction and Scalable Hypothesis algorithm was employed for the purpose of collecting and selecting the most optimal and pertinent temporal, statistical, and spectral features. The method achieved an accuracy and precision of 93.4% and 95%, respectively. When the investigation was repeated with RF, the accuracy achieved was 87%, in contrast to other supervised techniques like extreme gradient boosting, DT classification, light gradient boosting, and AdaBoost. These methods all yield accuracy results between 81% and 91%. In the beginning phase, the authors utilised data class

balancing techniques to enhance the dependability of their results. However, it did not have a substantial impact on accuracy, recall, or precision.

Hussain et al. have also introduced a sophisticated machine-learning framework dubbed NGBoost for NTL detection in [27]. A time-series feature-capturing tool known as Time Series Feature Extraction Library and the whale optimisation technique were used to produce statistical, temporal, and spectral attributes. In the aspect of effectiveness, the NGBoost technique accurately categorised consumers into either "Healthy" or "Theft" groups, surpassing the classification performance of RF, CATBoost classifier, AdaBoost classifier, DT classifier, and gradient boosting classifiers.

Khan I. et al. [28] classified individuals within the SGCC dataset as either "truthful" or "deceptive" using Bayesian SVM. Using a Bayesian optimisation algorithm, the classification model obtains a 94.1% level of accuracy, outperforming RF, Logistic Regression (LR) and SVM algorithms while also enhancing hyper-parameter tuning and the accuracy of the learning process. In [29], a comparison was made between different supervised ML algorithms, including DT, ANN, Deep Neural Networks (DNN), and Adaboost, using the SGCC dataset. The DNN algorithm stood out among the competing algorithms by demonstrating exceptional performance, achieving an accuracy rate of 93.04%. Nevertheless, the outcomes obtained need more reliability due to the inadequate performance observed in the recall and F1-score measurements. In [21], a different approach utilising a technique known as Extra Gradient Boosting (FA-XGBoost) was introduced and implemented on the SGCC dataset. A tool called the Visual Geometry Group was employed for characteristic retrieval, although the study did not emphasise its extra load [30]. Advanced techniques such as SVM, CNN, and LR were also evaluated to conduct a comparative analysis. The implemented system achieved a remarkable accuracy of 95%. For the purpose of grouping, the author's methodology introduces an approach that relies on a supervised ML algorithm referred to as Distributed Random Forest (DRF).

3 | METHODS AND MATERIALS

The presented technique (Figure 1) involves a two-stage process for extracting features. The first stage is characterised by data analysis, pre-processing, and feature extraction from the SGCC dataset. The analysis of the data is based on the idea that anomalous surges in the mean power consumption of consumers are probably associated with fraudulent activity. For example, Figure 2 illustrates that the recorded smart meter readings immediately transition to lower usage readings after a certain observed pattern. Similarly, if the disparity in usage is computed relative to an adjacent metre, a sudden upsurge in the neighbouring smart meter's readings becomes evident (Figure 3). In this stage, the dataset undergoes pre-processing to derive a collection of features from every smart meter reading. These features are derived by analysing a 2-month data window: a month preceding (lag) and a month following (lead)

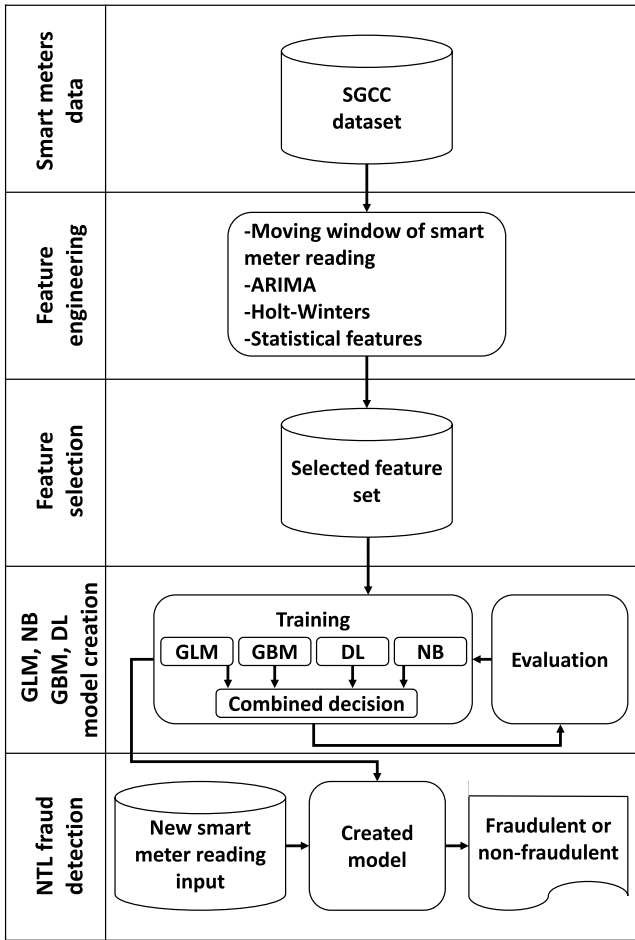


FIGURE 1 Illustration of the suggested non-technical loss (NTL) fraud detection approach.

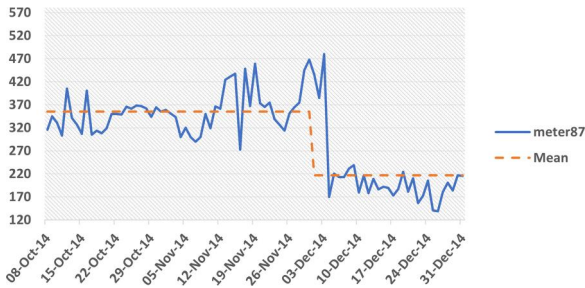


FIGURE 2 The attacker metre experienced an unexpected decrease in median consumption, indicating a significant drop in the readings of the smart meter. Blue is the used amount of smart meters. During the fraudulent moment, there is a spike (in orange) in the average value for the lagging metre measurements and the average for the leading metre measurements.

each reading. The finite delta differences are calculated using these readings, comprising the initial set of features. Next, an examination of the patterns evident in the moving mean of smart meter readings is performed, employing optimised moving mean methodology, including seasonality analysis,

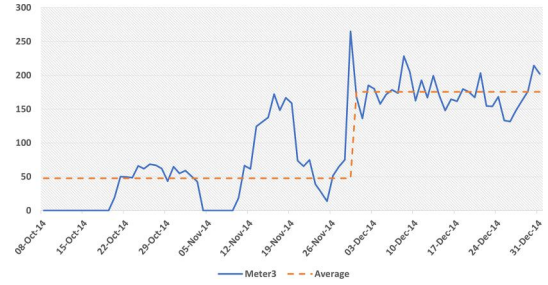


FIGURE 3 The recognition of an unexpected rise in the average usage of the compromised smart meter. The amount of usage recorded by the smart meter is represented by the blue line and the average of the lagging and leading smart meter readings is depicted in orange.

Holt-Winters, ARIMA, and usage trend. This analysis results in the generation of the second set of features.

3.1 | Feature engineering

First, a new set of features was engineered using the features obtained from a sliding window of leading and lagging readings of a normalised SGCC dataset with the related statistical features. The dataset was normalised by applying the min-max normalisation Equation (1) within the range of [0,1]. The building of the new feature sets and the statistical characteristics are outlined in the coming sections.

$$Norm = \frac{z - \min(z)}{\max(z) - \min(z)} \quad (1)$$

where z represents the initial feature value, $Norm$ denotes the standardised value, $\max(z)$ is the maximum value of z , and $\min(z)$ is minimum value of z .

3.1.1 | The identification of abrupt surges in smart meter consumption

To identify probable fraud, the proposed approach detects sudden changes in behaviour by generating a dataset comprising the attributes for each of the 61-m readings in a sliding window. This dataset, illustrated in Figures 2 and 3, encompasses a collection of derived features that characterise the temporal pattern surrounding each reading. These features include metrics like the pre-point and post-point means, the pre-point and post-point medians, the pre-point and post-point variances encompassing a range of 32 readings before and after, and thus the sum of finite differences (2).

$$\max \left(\sum_{i=1}^N (\delta(i)) \right) = \max \left(\sum_{j=1}^N (Lead_j - Lag_j) \right) \quad (2)$$

Where N is half the window size, Lag_j is the j th Lag reading, and $Lead_j$ is the j th Lead reading.

3.1.2 | Retrieval of characteristics from smart meter data

After determining unanticipated jumps, an additional batch of statistical characteristics is added to the time series data collection. Table 1 sums up and illustrates these statistical characteristics. Upon transforming the smart data into a series of fixed values (constant average and standard deviation), characteristics such as ARIMA, Holt-Winters, we can examine many characteristics, such as ARIMA, Holt-winters, trend, and seasonality. The ARIMA model is typically favoured for enhancing prediction accuracy, as it leverages time series data to forecast future trends based on historical values.

3.2 | Feature selection

Quite popular for feature selection, the *Boruta* algorithm uses random variables to determine which features consistently outperform random permutations of the original data. Gradient Boosting Machine is somewhat resistant to random features because it is an iterative algorithm that repeatedly re-scores the same data points, especially when used with row sampling. Later iterations can rectify a bad split-second decision. Observing the behaviour or random variables prevents poor divisions on other features.

3.3 | Classifications

As stated previously, four distinct methods, RF, GBM, and DL, are used in this study. The ML algorithms are employed on the SGCC dataset to classify smart meter results as either fraudulent or non-fraudulent. The results were compared to the results of various methods reported in literature.

TABLE 1 A collection of attributes specifically crafted for the purpose of detecting fraudulent activities.

Generated feature	Definition
Smart meter value moving window	Smart meter time series data reading values (32 leading + current + 32 lagging)
General statistic features	Sum, average, median, Variance, M
δ_1 to δ_{32} seasonality, randomness, trend	$\delta(i) = \text{Lead}(i) - \text{lag}(i)$ results of seasonal trend-decomposition.
Sum of finite differences	Delta(i) sum for current value neighbouring time series data
Holt-winters	Exponential averaging technique with the capacity to disregard unrelated readings
ARIMA	Autoregressive integrated moving Average
Label	1 for fraud or 0 for non-fraud label

3.3.1 | Generalised Linear Model

One popular ML algorithm used in detecting fraudulent smart meters is GLM. It is a flexible and highly useful statistical model-building method, with demonstrated effectiveness in identifying fraudulent activity by recording complicated connections among explanatory variables and the probability of fraud occurrence. Generalised Linear Model also provides extensive functionality with configurable hyperparameters, including selecting a type of distribution (Gaussian, Poisson, Binomial), the choice of link function (identity, log, logit, inverse), methods for regularisation (alpha and lambda parameters), and the treatment of missing values [31]. By leveraging the adaptability and capability of GLMs and optimising these hyperparameters, this research aims to identify deception in smart electricity metres efficiently, ensuring accurate analysis and maintaining the integrity of energy consumption data.

3.3.2 | Gradient Boosting Machine

The GBM is an effective and commonly utilised ML algorithm that is known for its capacity to manage complicated datasets and provide high predictive accuracy. It sequentially integrates an ensemble of weak prediction models, typically DT [32]. Each successive model in the group concentrates on correcting the mistakes of its predecessors, thereby enhancing the performance of the ensemble as a whole. Due to its ability to identify complex patterns and relationships within the data, GBM has been effectively applied to a variety of domains, including fraud detection.

Gradient Boosting Machine is one of the ML approaches evaluated in this study. We can optimise the effectiveness of the GBM model by adjusting its hyper-parameters, including its learning rate, tree count, and maximal tree depth. The learning rate dictates the role of each tree in the ensemble, whereas the number of trees and maximal tree depth govern the complexity and robustness of the model [33]. Through the assessment of GBM and its hyperparameters, we hope to determine its accuracy in identifying instances of fraudulent activity in electricity consumption data, thereby contributing to the development of strong fraud detection systems in smart meter networks.

The GBM model underwent testing twice with two different numbers of trees. Initially, the testing was done with the standard configuration, which employs 50 trees. The test was repeated with an increased number of 500 trees (Table 2). Additionally, stopping parameters were introduced to prevent overfitting when implementing early stopping settings. Three hyperparameters were defined as follows: *stopping metric* was used as the stopping criterion and was set to *Area Under Curve (AUC)*, *score tree interval* represented the frequency at which the model's performance was assessed every five trees, and *stopping rounds* terminated the training after completing three rounds.

TABLE 2 Gradient Boosting Machine (GBM) hyperparameters.

Hyper-parameter	Values
Number of trees	500
Batch size	25
Stopping rounds	3
Score tree interval	5
Stopping tolerance	0.0005
Stopping metric	AUC

3.3.3 | Deep Learning

A sub-field of ML, DL involves training ANN to learn hierarchical data representations automatically [34–37]. This enables DL models to capture intricate patterns and inter-dependencies within big datasets. In the context of recognising fraud in digital electricity metres, DL offers a number of advantages. It can manage high-dimensional and nonlinear data, making it suitable for analysing a variety of energy consumption patterns and characteristics. Moreover, DL algorithms have remarkable generalisation capabilities, allowing them to adapt to different fraud patterns and detect anomalies that may elude traditional rule-based or statistical methods.

As it is one of the ML methods often considered in literature, DL is also investigated in this study. Deep Learning is a powerful method known to employ neural networks via numerous layers in order to gather complicated patterns as well as characteristics from data. The investigation has examined DL's potential in fraud detection by capitalising on its capacity to manage high-dimensional and nonlinear data. The structure of the neural network, including its total amount of layers that are concealed as well as the amount of nodes inside every layer, can also be taken into account as hyperparameters. Moreover, hyperparameters associated with regularisation, including *dropout rate* and *weight decay*, could further be studied to increase the model's ability to generalise and avoid overfitting.

The DL machine was tested three times with two different sets of epoch numbers. Firstly, it was evaluated with the initial default settings of “10” epochs and since the early stopping parameter is enabled by default, the default stopping parameters were used to perform early stopping. The experiment was then repeated with the epoch number set to “20”. For comparison, early stopping was disabled by setting *stopping rounds* to “0”. For the third experiment, we used the same model parameters as *dl fit2* but early stopping was enabled and the stopping criteria were specified. We also passed a validation set, as it is recommended for early stopping and is valid only if *stopping rounds* is greater than “0”. The hyperparameters for *dl fit2* are given in Table 3, and the hyperparameters for *dl fit3* are given in Table 4. Moreover, it is feasible to enhance the model's performance by increasing the number of epochs in a deep neural network. Once the optimal parameter values were

TABLE 3 Deep Learning (DL) hyperparameter (*dl fit2*).

Hyper-parameter	Values
Epochs	20
Hidden	C (10,10)
Stopping rounds	0

TABLE 4 Deep Learning (DL) hyperparameter (*dl fit3*).

Hyper-parameter	Values
Epochs	20
Hidden	C (10,10)
Score interval	1
Stopping_rounds	3
Stopping_metric	AUC
Stopping_tolerance	0.0005

obtained, the model was trained and evaluated using the DL algorithm over the SGCC time-series data.

3.3.4 | Naive Bayes

Naive Bayes is a classification algorithm that is usually used as an alternative to DT. It is characterised by applying Bayes Theorem while strongly assuming independence of covariates. Explicitly, the algorithm assumes that predictor variables are independent of each other given the response. Furthermore, this method assumes that the numeric predictors adhere to a Gaussian distribution, with their standard deviations and means obtained by statistical computations involving the training dataset. During the construction of a NB classifier, a row that contains at least one missing value in the training dataset is entirely excluded. In the event of a test dataset with missing values, those particular predictors will be omitted from the probability calculation during the prediction process.

The NB classifier algorithm is one of the ML algorithms that is investigated in this study. As with the other algorithms, hyperparameters can be configured to give different operating conditions for the models. Hypothetically, this may give different results. One of the configurable hyperparameters is the amount of *Laplace smoothing*. Due to the fundamental nature of the NB algorithm, it may not perform well in real-world scenarios by default. Laplace smoothing is a solution employed as a smoothing technique to tackle the issue of zero probability.

The NB algorithm was tested two times with the main difference between the two experiments being the amount of Laplace smoothing. By default, the NB model does not use any Laplace smoothing. First, a NB model was trained using default parameters. For the second experiment, another NB model was trained using Laplace smoothing, using the parameter *laplace* set to “6”. After the parameters were set and training done, the models were evaluated using the SGCC dataset.

4 | EXPERIMENTAL EVALUATION

4.1 | Dataset preparation

The introduced model ability to detect theft and fraud from electricity smart meters readings is verified over the SGCC dataset [2]. The SGCC dataset comprises electrical energy smart meter readings for over 32,000 users during a 1035-day time frame. This dataset undergoes a two-stage categorisation procedure. A subset of 7000 entries is used for the investigation, of which 520 are identified as fraudulent users and 6480 as legitimate users. The 7000 m (520 fraudulent vs. 6480 non-fraudulent), each with 1035 readings, have been transformed into a total of 7,245,000 records in the transformed features. Out of these, 538,200 records are fraudulent and 6,706,800 are non-fraudulent. Further details are discussed in section 6.3.

4.2 | Performance measures

The performance of the ML algorithms was evaluated using the performance measures detailed in Table 5. In the equations defining some of the performance measures, *FP*, *TP*, *FN*, and *TN* represent False Positive, True Positive, False Negative, and True Negative, respectively. Also, *AUC* stands for Area Under the Curve, *MSE* is the Mean Square Error, *RMSE* represents the Root Mean Square Error, and \bar{y} signifies the mean of *y* values or the expected value of *y*.

5 | RESULTS

In this work, we divided the engineered dataset into two subsets for training and validation (or testing) purposes. We implemented four ML algorithms using the programming language *R*. For processing and transforming the dataset, we used the ‘*dplyr*’ library. Statistical feature calculation was done

using the ‘*matrixStats*’ library. Auto-Regressive Integrated Moving Average feature extraction was performed using the ‘*Forecast*’ library, and for ML algorithms and data visualisation, we relied on ‘*tidyverse*’, ‘*H₂O*’ [38], and ‘*ggplot2*’. The ML models were trained on a system with a core-i7 processor, 32 GB RAM, and a 12 GB NVIDIA GTX graphics card. Detailed results of each ML algorithm are presented in the following subsections.

5.1 | Generalised Linear Model results

The results of the two experiments with the GLM algorithm show that it achieves a sensitivity (recall) score of 1.0 for both the experiments. In the first experiment, the accuracy achieved was 0.999,782, both the specificity and precision scores were 0.999,995, the obtained F1-score was 0.999,783, and the obtained AUC was 0.9999995. Similarly, the second experiment gave an accuracy score of 0.999,801, while it achieved specificity and precision scores of 1.0, an F1-score of 0.999,801, and an AUC that was 0.9999989. A summary of the results of the experiments with GLM is shown in Tables 6 and 7. The graphical representation of the results in those tables can be seen in Figure 4.

5.2 | Gradient Boosting Machine results

The two experiments with GBM have shown interesting results. The GBM models in the experiments performed amicably and gave scores of 1.0 for all performance measures (Acc, Sn, Sp, Pr, F1, AUC). There were few variations in the model losses and R^2 values; however, the values are consistent with the overall performance of the models. In the first experiment, the MSE was 0.000,010,569, the RMSE was 0.003,251,006, the Logloss was 0.00,325,624, and the R^2 value was 0.9999577. The model loss scores in the second

TABLE 5 Metrics used to evaluate the performance of the various employed machine learning (ML) methods.

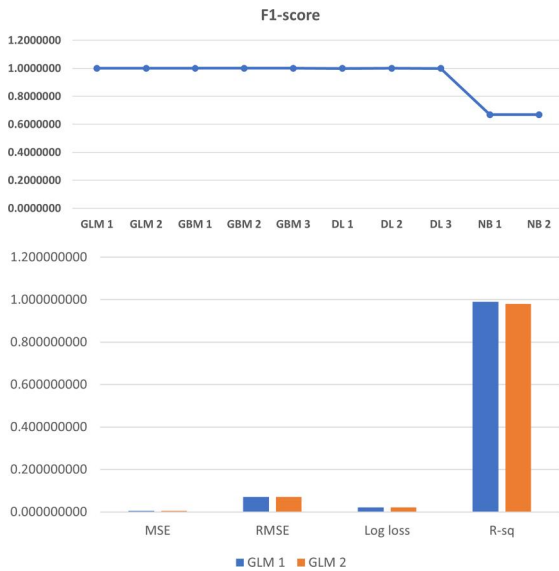
Performance measure	Definition	Formula
Accuracy (Acc)	Measure of how frequently both fraudulent and non-fraudulent instances are correctly identified.	$Acc = \frac{(TP+TN)}{(TP+FP+FN+TN)}$
Recall or sensitivity (Sn)	Determines the method's accuracy when identifying fraud.	$Sensitivity = \frac{TP}{(TP+FN)}$
Specificity (Sp)	Determines the method's accuracy when identifying non-fraud.	$Specificity = \frac{TN}{(TN+FP)}$
AUC	Area under the curve.	$AUC = \frac{Rank_{i \text{ positive class}}}{M \times N}$
Precision (Pr)	Evaluates the classifier's ability to accurately distinguish instances of fraud from cases of actual fraud.	$Pr = \frac{TP}{(TP+FP)}$
F1-score	Recall and precision are harmonically averaged to give the F1 score.	$F1score = 2 \times \frac{(Pr \times Sp)}{((Pr+Sp))}$
MSE	The average square of the discrepancy between the estimated & actual values	$MSE = \frac{1}{n} \sum_{i=1}^n (\bar{y}_i - y_i)^2$
RMSE	The square root of MSE	$RMSE = \sqrt{\frac{\sum_{i=1}^n (\bar{y}_i - y_i)^2}{n}}$
Logloss	The cross-entropy loss	$LogLoss = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^M x_{ij} \log(p_{ij})$
R^2	The measure of the proportion of the variance in a dependent variable explained by an independent variable in regression analysis	$R^2 = 1 - \frac{\sum_{i=1}^n (\bar{y}_i - y_i)^2}{\sum_{i=1}^n (\bar{y}_i - \bar{y})^2}$

TABLE 6 Generalised Linear Model (GLM) performance results.

Exp	Acc	Sn	Sp	Pr	F1	AUC
GLM 1	0.999,782	1.0	0.999,995	0.999,995	0.999,783	1.0
GLM 2	0.999,801	1.0	1.0	1.0	0.999,801	1.0

TABLE 7 Generalised Linear Model (GLM) model losses.

Exp	MSE	RMSE	Logloss	R^2
GLM 1	0.004,925,593	0.070,182,570	0.021,145,890	0.989,297,300
GLM 2	0.004,900,019	0.070,000,130	0.021,066,740	0.980,399,600

**FIGURE 4** Results of Generalised Linear Model (GLM) experiments. The top sub-figure shows the performance measures and the bottom shows the model losses.

experiment were 0.0 and the R^2 obtained was 1.0, indicating a higher performance than in experiment 1 and a better-performing model overall. The model losses are organised in Table 8. The graphical illustrations of the same results are shown in Figure 5.

5.3 | Deep Learning results

There were three experiments conducted using DL. For experiment 1, the model performed well with an accuracy of 0.999,283, a sensitivity score of 1.0, specificity and precision scores of 0.999,937, an obtained F1-score of 0.999,286, and 0.9,998,998837 as the AUC. The model losses were as follows: MSE - 0.000,685,124, RMSE - 0.02,617,488, and logloss - 0.005,926,242.

In the second experiment, the model performed similar to, if not better than experiment 1 in many measures. The accuracy obtained was 0.999,844, sensitivity was 1.0, the specificity was 0.999,912, this was close to the precision which was 0.999,913, the obtained F1-score was 0.999,845, and the AUC

TABLE 8 Gradient Boosting Machine (GBM) model losses.

Exp	MSE	RMSE	Logloss	R^2
GBM 1	0.000010569	0.003,251,006	0.003,256,240	0.999,957,700
GBM 2	0.000000000	0.000000000	0.000000000	1.000000000
GBM 3	0.000578,289	0.024,047,640	0.024,341,050	0.997,686,800

**FIGURE 5** Results of Gradient Boosting Machine (GBM) experiments. Performance measures are shown in the top sub-figure and the model losses are shown in the bottom sub-figure.

obtained was 0.9999303. In terms of model losses, the model gave comparatively lower values of each error measure. The MSE was scored at 0.000,154,854, the RMSE at 0.01,244,404, and the logloss at 0.002,856,985.

It could be argued that the model in the third experiment performed less effectively compared to the previous two experiments. The overall performance results are high nonetheless. The accuracy obtained in this trial was 0.999,395, while the sensitivity was 1.0 again, the specificity and precision scores were 0.999,486. An F1-score of 0.999,398 was obtained, and an AUC of 0.9,997,067 was observed. The model losses were as follows: MSE - 0.000,952,688, RMSE - 0.03,086,564, logloss - 0.008,014,885. It is worth noting that the R^2 values could not be obtained in the experiments with DL. Tables 9 and 10 of results for the three experiments. Figure 6 illustrates those results graphically.

5.4 | Naive Bayes results

The last ML algorithm tested was the NB classifier. The results for the two experiments conducted using this algorithm are identical. This indicates that the parameters set for the models

TABLE 9 Deep Learning (DL) performance results.

Exp	Acc	Sn	Sp	Pr	F1	AUC
DL 1	0.999,283	1.0	0.999,937	0.999,937	0.999,286	0.999,884
DL 2	0.999,844	1.0	0.999,912	0.999,913	0.999,845	0.999,930
DL 3	0.999,395	1.0	0.999,486	0.999,486	0.999,398	0.999,707

TABLE 10 Deep Learning (DL) model losses.

Exp	MSE	RMSE	Logloss
DL 1	0.000685,124	0.026,174,880	0.005,926,242
DL 2	0.000154,854	0.012,444,040	0.002,856,985
DL 3	0.000952,688	0.030,865,640	0.008,014,885

**FIGURE 6** Results of Deep Learning (DL) experiments. Performance measures are shown at the top, followed by the model losses.

have no effect on the algorithm's performance. In both cases, the classifier scored 0.52 in terms of accuracy. The achieved scores for sensitivity, specificity, and precision were 1.0, 0.986,105, and 0.796,137, respectively. An F1-score of 0.668,538 was obtained, while the AUC was 0.5,219,506. The recorded MSE was 0.4,811,734, the RMSE was 0.6,936,667, and the logloss was rather high at 16.55,273. The scores of the performance measures are shown in Table 11. Table 12 shows the model losses. Figure 7 shows the graph of the results.

6 | Discussion of results

The objective of the experiments was to determine the best-performing ML algorithm with the SGCC dataset. This falls in line with the expected outcomes, as mentioned in Section 1,

TABLE 11 Naive Bayes (NB) performance results.

Exp	Acc	Sn	Sp	Pr	F1	AUC
NB 1	0.52	1.00	0.986,105	0.796,137	0.6,685,380	0.5,219,506
NB 2	0.52	1.00	0.986,105	0.796,137	0.6,685,380	0.5,219,506

TABLE 12 Naive Bayes (NB) model losses.

Exp	MSE	RMSE	Logloss
NB 1	0.481,173,400	0.693,666,700	16.552,730,000
NB 2	0.481,173,400	0.693,666,700	16.552,730,000

**FIGURE 7** Results of Naive Bayes (NB) experiments with performance measures (top) and model losses (bottom).

and the goal of introducing a full-fledged, high-performing NTL fraud detection solution. Four ML algorithms were evaluated: GLM, GBM, DL, and NB classifiers. As mentioned in the previous section, the dataset was divided into two sets for training and evaluation. Experiments were done using the R programming language with the libraries and settings mentioned previously. Figures 8 and 9 show the graphical comparison of all the models tested in terms of each performance and model loss measure. The results show that the highest-performing ML algorithm on the SGCC dataset is GBM. In particular, the second GBM experiment yielded the highest scores for the performance measures in conjunction with being the lowest-scoring trial in terms of model losses. Therefore, we propose the GBM model as an effective and accurate classifier for the purposes of electricity and NTL fraud detection.

The results also show that all the models tested, except the NB classifier, performed amicably in many areas. The two experiments with the NB classifier yielded identical results. This

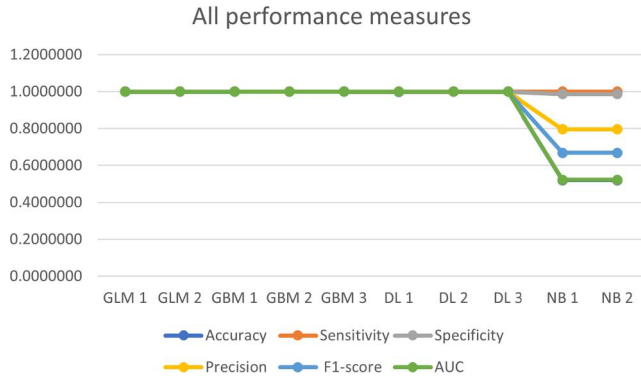


FIGURE 8 Comparison of results of the performance metrics: Accuracy (blue), Sensitivity (orange), Specificity (grey), Precision (amber), F1-score (light blue), Area Under Curve (AUC) (green).

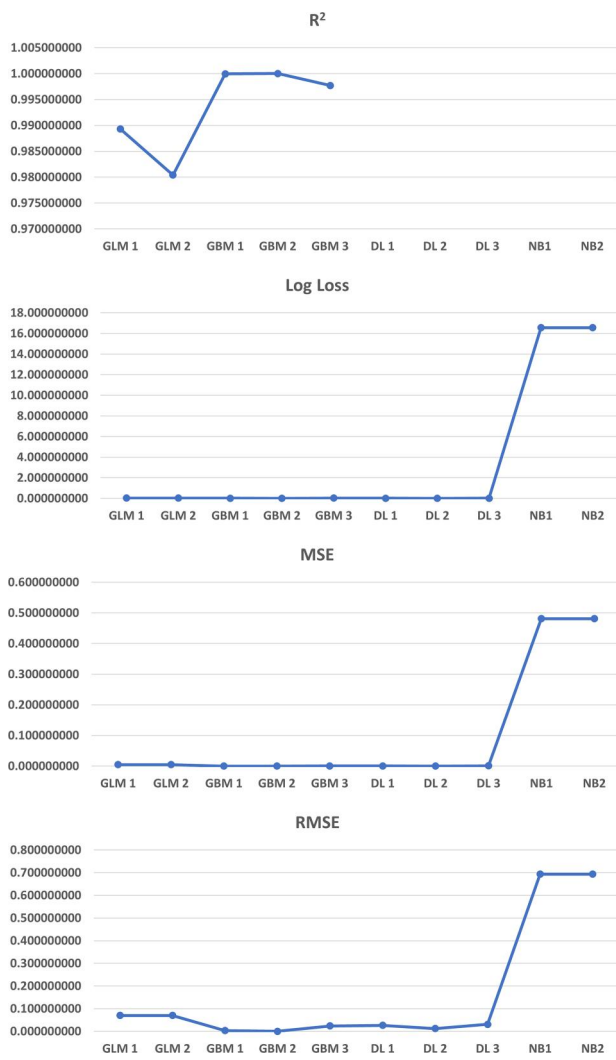


FIGURE 9 The sub-figures show the R^2 , Logloss, MSE, and RMSE scores of all the tested algorithms. The higher the R^2 the better, and the lower the values of Logloss, MSE, and RMSE the better.

clearly shows that the parameters set for the two experiments do not affect the performance of the algorithm with this dataset. One of the few tuneable model parameters for the NB algorithm

is the amount of Laplace smoothing. By default, the ‘ H_2O ’ NB models do not use any Laplace smoothing. In experiment 1, default settings were used, while the model was trained with Laplace smoothing in the second experiment. It follows, therefore, that Laplace smoothing does not influence the performance of the NB classifier with the dataset used. In both experiments, the algorithm scored 0.5,200,000 in terms of accuracy, a 1.0 for sensitivity, 0.9,861,050 for specificity, 0.7,961,370 for precision, 0.6,685,380 as the F1-score, and 0.5,219,506 as the AUC. These scores are lower than the performance scores of the other algorithms tested. The algorithm also gave higher model losses than the other algorithms, as can be seen in Figure 9. This leads to the conclusion that, overall, the NB algorithm does not perform well with the used dataset. In this section, a discussion follows on the performance of the proposed ML algorithm in terms of the evaluation criteria, or metrics mentioned (performance measures and model losses). Additionally, the results are analysed to infer a comparison between the training and validation of the proposed model, thereby illustrating how well the model fits the SGCC dataset.

6.1 | Gradient Boosting Machine algorithm advantages

One main advantage of the proposed solution is the ability to extract the finite differences as features in the transformed dataset and the power the GBM algorithm has in detecting the finite differences gathered in the pre-processing stage. The results of the GBM experiments are consistent with the expected performance of the algorithm. It is often considered a type of gradient descent algorithm. As such, there are numerous advantages and disadvantages to the GBM algorithm. The first of the many advantages is that high accuracy in predictions is more easily attainable with GBM. Secondly, adjusting and optimising hyperparameters for various loss functions makes the fit of the function more adaptable. Moreover, GBM performs admirably with numerical and categorical values without requiring pre-processing. Lastly, missing data can be accounted for without involving imputation. At the other end of the spectrum, enhanced GBM continues to minimise errors, leading to overemphasising and overfitting outliers. To neutralise this, cross-validation is required. Furthermore, GBM models require a significant number of trees, which can be memory- and time-intensive. Lastly, numerous variables (iterations, regularisation parameters, tree depth, and so forth) influence the behaviour of the approach owing to its adaptability. This requires an extensive grid search when tuning. While less interpretable, LIME, partial dependency graphs, variable worth, and similar tools may be of assistance.

One advantage of GBM is its ability to tune hyperparameters. Perhaps the only drawback to this is that tuning is time-intensive. The most commonly tuned hyperparameters include the number of trees, depth of trees, learning rate, and subsampling. In this study, the number of trees was tuned, and the rest of the hyperparameters were left unchanged.

6.2 | Best number of trees

It is worth noting that the number of DT affect the performance of the GBM algorithm. In the first GBM experiment, default parameters were used and, hence, the model was trained using 50 trees, which is the default number. In the second experiment, the number of trees was increased to five hundred. However, no early stopping parameters were set in these two experiments. To avoid overfitting due to the large number of trees, early stopping was enabled, and the related parameters were set in the third experiment, keeping the same number of trees as the previous trial. Another important point to note is that early stopping is not enabled by default for the GBM algorithm. Only DL, out of the four tested ML algorithms, has early stopping available by default.

Since the GBM algorithm scored perfect scores for the performance measures in all the three experiments, the model losses must be compared to evaluate the overall effectiveness of the models. Despite the fact that the losses are very minimal, there is a variation in the values for the three experiments owing to the different parameters, namely the number of trees and early stopping parameters. From the results in Table 8 and Figure 5, it can be easily inferred that the model parameters of the second experiment give the best overall results as there are no model losses and the R^2 value is 1.0. The second best results are those of experiment 1, followed by those of the third experiment.

The reason that experiment 3 had the highest model losses out of the three experiments, despite having the number of trees set to '500' like experiment 2, is the effect of early stopping on the training. The three parameters used to control early stopping are 'stopping rounds', 'stopping metric', and 'stopping tolerance'. Another parameter which is used, but does not influence early stopping, is 'score tree interval'. The 'stopping rounds' parameter determines the number of iterations that must be completed until the 'stopping tolerance' of the 'stopping metric' is crossed. Setting a 'score tree interval' scores the model for that many number of intervals.

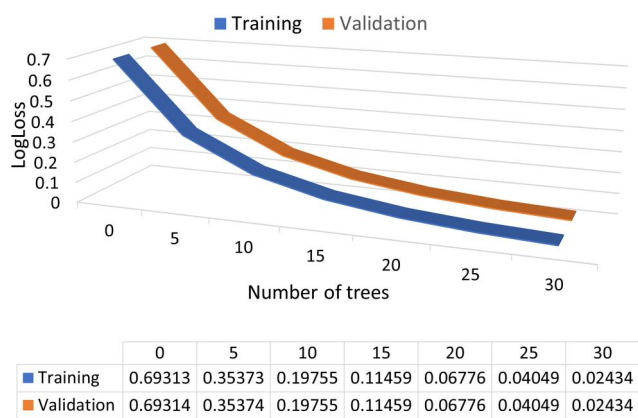


FIGURE 10 The increase in the number of trees minimises the training and evaluation model loss.

The 'stopping metric' is the metric by which performance can be measured; it was set to AUC. The 'stopping rounds' parameter was set to 3, the 'stopping tolerance' to 0.0005, and the 'score tree interval' to 5. These hyperparameters are given in Table 2.

As a result of the early stopping conditions, the model only trained with 30 trees, since a perfect 1.0 score for the AUC had been achieved and, therefore, it remained unchanged for 3 rounds. This was enough for the set conditions to be met. Moreover, the number of trees ultimately used is less than the default 50 trees used in experiment 1, hence the trend in the model losses between the experiments. It must be noted that the effect is minimal, and the model losses in Experiment 3 are not much greater than those in Experiment 1. Despite the difference in model losses between the three experiments, experiment 3 was trained much faster than the other two. This is an advantage, as the higher number of trees in experiment 2 increases the computational burden and, hence, the training time of the model. Thus, there is a trade-off between reducing the losses and training time. Therefore, the conclusion that can be appropriately drawn is that increasing the number of DT decreases the model losses (Figure 10), at the expense of increased computational burden and longer training time. Figure 11 illustrates the scoring history of the second GBM experiment and clearly shows that the model losses decrease as the number of trees increases. It can be observed that the model was perfectly trained, and there are no errors as the training and validation curves overlap. Figure 12 shows the same results for the DL experiments for comparison.

6.3 | Class imbalance handling

Although the used dataset has a class imbalance issue, the reported results have handled the class imbalance in all the experiments as follows: When 'balance classes' is enabled, H2O may undersample the majority or oversample the minority classes. Enabling the balance class option increased the data frame size. It provided the parameter 'max after balance size' that managed to reduce the data frame size. This specifies the maximum relative size of the input dataset after balancing class counts and defaults to 5.0.

The under-sampling technique is applied to mitigate the issue of class imbalance. In the third GBM experiment (Table 13), for example, undersampling is implemented by selecting 160,302 records as undersampled features from the given dataset. Out of these, 79,813 records are classified as fraud and 80,489 records are classified as non-fraud. The confusion matrices (Tables 13–16) for the methods clearly show that the work has employed the undersampling or weighting methods to address class imbalance and assure the reliability of the results. The confusion matrix for the third GBM experiment shows that the ratio of class 0 over class 1 is 1:1, which means the class imbalance is handled by under-sampling the majority class to be balanced with the minority class.

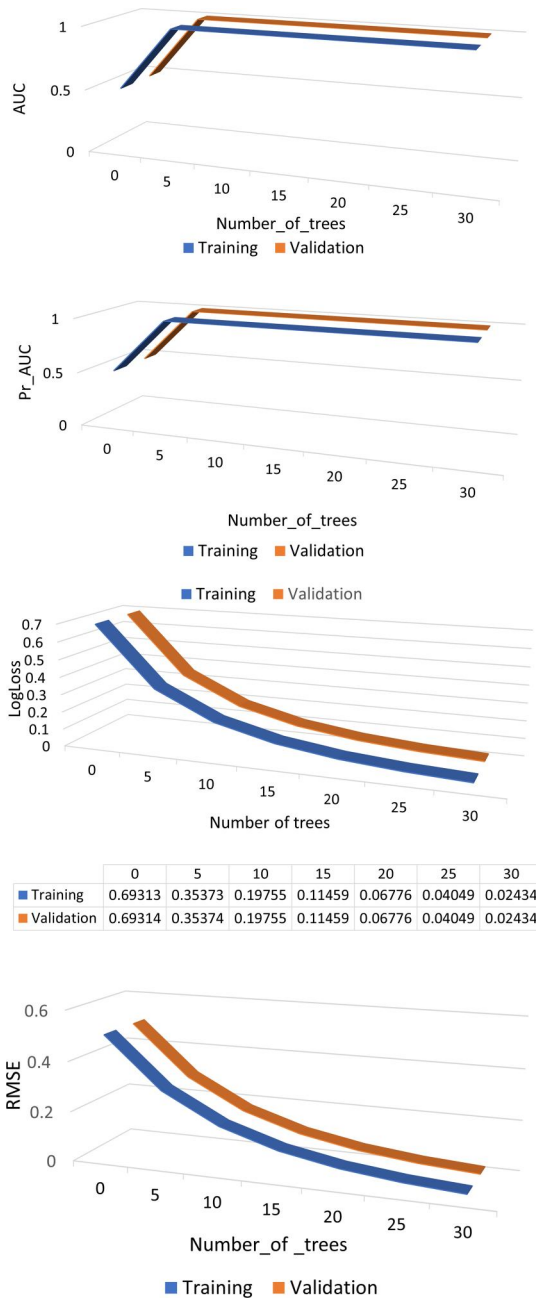


FIGURE 11 The vertically arranged sub-figures show the *AUC*, *PR_AUC*, *Logloss*, and *RMSE* results for the Gradient Boosting Machine (GBM) training and validation models with respect to the number of trees.

6.4 | Ensemble learning

GBMs are members of the family of ensemble learning techniques. This work shows the high performance of the GBM model in six performance measures and the adaptability of the proposed powerful NTL fraud detection model. Since it is an ensemble method, three more experiments were performed using DRF on the transformed SGCC dataset with different parameters. In the first experiment, default parameters are set. In the default configuration, the algorithm utilises 50 trees. In the second experiment, the number of trees was increased to 100. Usually, increasing the number of trees increases

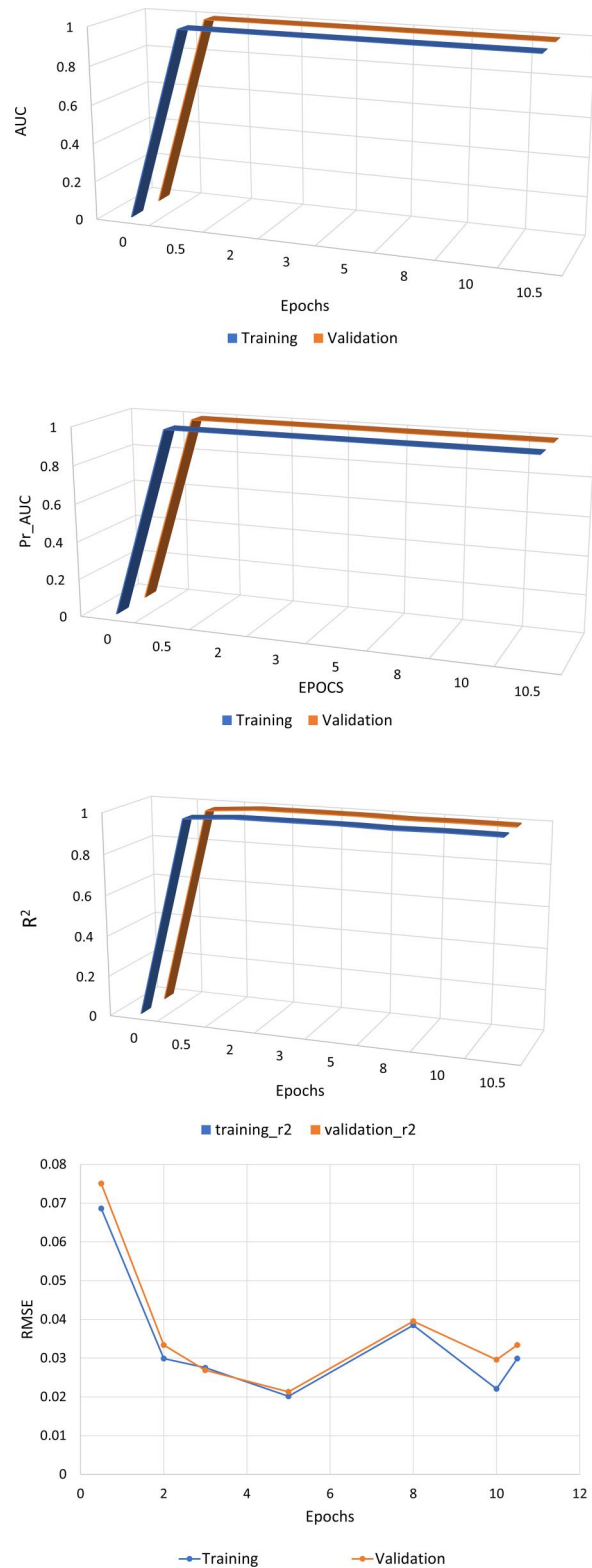


FIGURE 12 The vertically arranged sub-figures show the *AUC*, *PR_AUC*, *R²*, and *RMSE* results for the Deep Learning (DL) training and validation models with respect to Epoch number.

performance as well. It is also considered that RF algorithms are fairly resistant to overfitting. Table 17 shows a comparison of the model losses from GBM and RF experiments.

TABLE 13 Gradient Boosting Machine (GBM) 3 confusion matrix (vertical: actual; across: predicted).

	0	1	Error	Rate
0	79,813	0	0	0/79,813
1	0	80,489	0	0/80,489
Totals	79,813	80,489	0	0/160,302

TABLE 14 Generalised Linear Model (GLM) 2 confusion matrix (vertical: actual; across: predicted).

	0	1	Error	Rate
0	79,907	15	0.000188	15/79,922
1	17	80,573	0.000211	17/80,590
Totals	79,924	80,588	0.000199	32/160,512

TABLE 15 Deep Learning (DL) 2 confusion matrix (vertical: actual; across: predicted).

	0	1	Error	Rate
0	79,806	7	0.000088	7/79,813
1	18	80,471	0.000224	18/80,489
Totals	79,824	80,478	0.000156	25/160,302

TABLE 16 Naive Bayes (NB) 2 confusion matrix (vertical: actual; across: predicted).

	0	1	Error	Rate
0	0	79,813	1.000000	79,813/79,813
1	0	80,489	0.000000	0/80,489
Totals	0	160,302	0.497,891	79,813/160,302

TABLE 17 Comparison of model losses in Distributed Random Forest (DRF) and Gradient Boosting Machine (GBM) models.

Exp	MSE	RMSE	Logloss	R^2
DRF 1	0.001,350,698	0.03,675,185	0.03,127,834	0.9,945,971
DRF 2	0.001,648,786	0.04,060,525	0.03,570,996	0.9,934,047
GBM 1	0.000010569	0.003,251,006	0.00325,624	0.9,999,577
GBM 2	0.0	0.0	0.0	1.0
GBM 3	0.000578,289	0.02,404,764	0.02,434,105	0.9,976,868

Comparing the results presented in Table 17, we notice that R^2 values from the GBM experiments are higher than those from the DRF experiments. This indicates that the GBM model is more capable of representing the fraud behaviour. A higher R^2 value indicates how strong the model is in representing the studied phenomena.

6.5 | Comparison with other methods

Existing literature encompasses various methods and ML algorithms employed for the detection of electricity fraud and NTLs. When compared to other state-of-the-art methods, the proposed approach exhibits superior performance across all six tested performance measures. The six performance measures give a good indication of which algorithm performs better with a given dataset, especially for NTL detection.

Accuracy measures all the correctly classified instances overall. Recall measures how many cases are correctly classified as “true positive” (TP). Likewise, specificity measures how many cases were correctly classified as “true negative” (TN). Precision is the ratio of correctly classified TP cases to the actual number of TP cases. The F1-score is usually a measure of the balance between recall and specificity. Lastly, the AUC is a measure that is high if both recall and specificity are high.

Table 18 quantitatively compares the GBM algorithm and the state-of-the-art methods referenced in the literature. The results indicate that the proposed model outperforms the compared models across all performance measures, thereby demonstrating its efficacy in the detection of electricity fraud and NTLs. Figure 13 depicts the performance of the proposed method (shown in blue) and the other reported approaches, using the AUC benchmark. The proposed GBM model exhibited the largest AUC in both the Receiver Operating Characteristic AUC, which assesses the trade-off between sensitivity and specificity, and the area under the curve for precision and sensitivity (PR-AUC).

6.6 | Future work

This study was conducted on 4 ML models, with a total of 10 experiments (two to three experiments in each model), and involved the testing of 72 columns for feature selection (using the Boruta Algorithm). The goal was to select the best-contributing columns and drop the bad and constant columns, which ultimately resulted in a promising two-stage method for identifying NTLs in smart meters caused by electricity theft and fraud.

Further research work is planned in the future to, firstly, evaluate the model on a wider range of smart-grid datasets to demonstrate that the technique is effective for various smart-grid consumption data and to reduce its reliance on a specific dataset; secondly, to conduct ablation tests to measure the influence of the feature engineering phase and the selection of ML methods. In summary, this study presents a pragmatic and precise two-step approach to address this significant issue by using both feature engineering and advancements in the field of ML.

7 | CONCLUSION

Non-technical losses may arise from electricity fraud and are causing significant financial losses for utility companies, amounting to millions of dollars. This study introduces an

Method	Ref	Year	Sn	Sp	Pr	Acc	AUC	F1
Proposed method		2023	1.00	1.00	1.00	1.00	1.00	1.00
Random forest	[39]	2022	0.98	0.99	0.99	0.98	0.98	0.98
AdaBoost	[29]	2021	0.07		0.57	0.91	0.53	0.13
Ensemble NTL	[40]	2021	0.98		1.00		0.99	0.99
Deep ANN	[29]	2021	0.40		0.59	0.92	0.69	0.45
NGBoost	[27]	2021	0.91		0.95	0.93	0.94	0.92
ANN	[29]	2021	0.35		0.64	0.92	0.66	0.42
CatBoost	[26]	2021	0.92		0.95	0.93		0.94
Bayesian Support Vector Machine (BSVM)	[28]	2021	0.91		0.96	0.94	0.93	0.94
DERUSBOOST	[41]	2020	0.90	0.99	0.90	0.96	0.99	
CNNGRUPSO	[42]	2020				0.87	0.89	
FA-XGBoost	[21]	2020	0.97		0.93	0.95	0.95	0.94
Wide and Deep CNN (WADCNN)	[2]	2018	0.74	0.87	0.70	0.86	0.76	

TABLE 18 Comparison of our proposed method with the methods reported in literature.

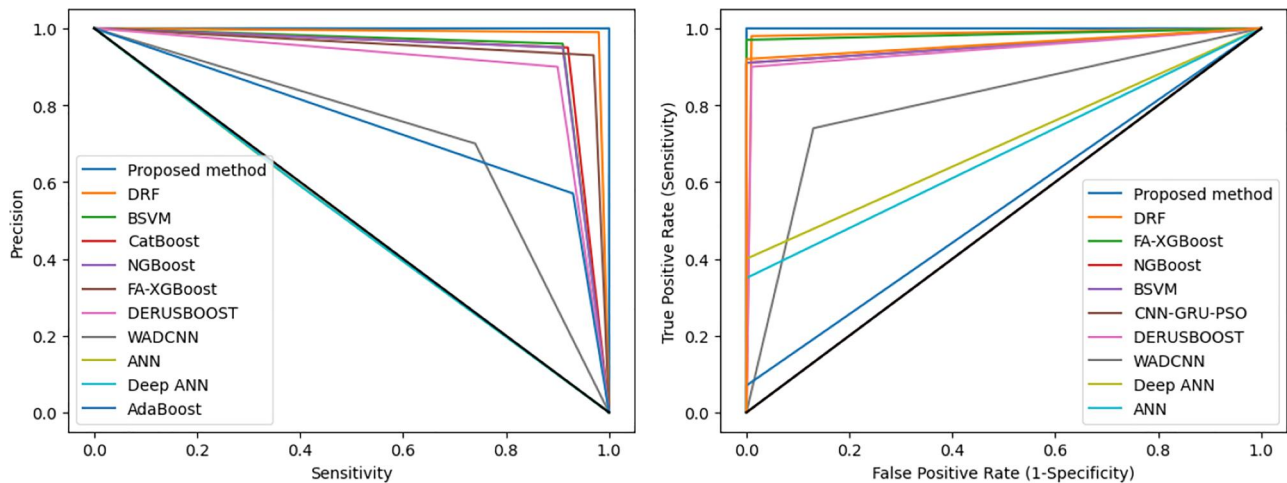


FIGURE 13 Performance of the proposed method compared with that of the state-of-the-art in terms of PR_AUC (left) and ROC_AUC (right).

innovative and pragmatic approach for identifying NTLs and electricity fraud in smart grids. The suggested GBM model exhibits a noteworthy characteristic in its ability to achieve performance metric scores of over 0.99, indicating its reliability and consistency. This approach involves the creation of several trees, each of which is trained and fitted differently. The decisions made are subsequently aggregated to get accurate results. The proposed methodology involves a two-step process. First, sudden jumps or abrupt changes are detected using the sum of finite differences around a specific point, and the process is enhanced through the use of ARIMA and Holt-Winters models. Following this, features are extracted from the smart meter readings in the SGCC dataset. The transformed data is fed to an ML algorithm, which classifies the metre as either fraudulent or non-fraudulent. The use of finite differences enables ML algorithms to detect sudden jumps or abrupt changes, and time series feature extraction transforms the dataset for better ML performance. The efficacy of the

method that has been proposed can be seen in the experiments that were conducted. The findings of this innovative approach demonstrate its increased effectiveness compared to existing approaches in terms of precision, accuracy, AUC, specificity, recall, and F1 score.

AUTHOR CONTRIBUTIONS

Sufian A. Badawi: Conceptualisation, Methodology Design, Software Design, Validation, Formal analysis, Investigation, Data curation, Writing the original draft, Writing, review and editing the final draft, Visualisation, Supervision. **Maen Takruri:** Conceptualisation, Methodology Design, Validation, Formal analysis, Investigation, Resources, Writing the original draft, Writing, review and editing the final draft, Supervision, Project administration. **Mahmood G. Al-Bashayreh:** Conceptualisation, Methodology Design, Validation, Formal analysis, Investigation, Writing, review and editing the final draft. **Khoulood Salameh:** Conceptualisation, Methodology Design,

Formal analysis, Investigation, Resources, Writing the original draft. **Jumana Humam**: Software Design, Formal analysis, Investigation, Data curation, Writing the original draft, Visualisation. **Samar Assaf**: Software Design, Formal analysis, Investigation, Data curation, Writing the original draft, Visualisation. **Mohammad R. Aziz**: Validation, Formal analysis, Investigation, Writing the original draft, Writing, review and editing the final draft. **Ameera Albadawi**: Software Design, Formal analysis, Data curation, Writing the original draft, Visualisation. **Djamel Guessoum**: Conceptualisation, Methodology Design, Formal analysis, Investigation, Writing, review and editing the final draft. **Isam ElBadawi**: Conceptualisation, Formal analysis, Investigation, Writing, review and editing the final draft. **Mohammad Al-Hattab**: Conceptualisation, Formal analysis, Investigation, Writing, review and editing the final draft.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

The source dataset of this study is available at [Electricity Theft Detection] GitHub site at <https://github.com/henryRDlab/ElectricityTheftDetection>.

ORCID

Maen Takeruri  <https://orcid.org/0000-0001-9785-3920>

Mohammad R. Aziz  <https://orcid.org/0009-0009-9827-1629>

Mohammad Al-Hattab  <https://orcid.org/0000-0001-8359-7428>

REFERENCES

- Haq, E., et al.: Electricity-theft detection for smart grid security using smart meter data: a deep-CNN based approach. *Energy Rep.* 9, 634–643 (2023). <https://doi.org/10.1016/j.egy.2022.11.072>
- Zheng, Z., et al.: Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids. *IEEE Trans. Ind. Inf.* 14(4), 1606–1615 (2018). <https://doi.org/10.1109/tii.2017.2785963>
- Hasan, M., et al.: Electricity theft detection in smart grid systems: a CNN-LSTM based approach. *Energies* 12(17), 3310 (2019). <https://doi.org/10.3390/en12173310>
- Nagi, J., et al.: Nontechnical loss detection for metered customers in power utility using support vector machines. *IEEE Trans. Power Deliv.* 25(2), 1162–1171 (2009). <https://doi.org/10.1109/tpwr.2009.2030890>
- Glauner, P., et al.: The challenge of non-technical loss detection using artificial intelligence: a survey. *arXiv*, 00626 (2016). [arXiv:1606.00626](https://arxiv.org/abs/1606.00626)
- Saeed, M.S., et al.: Detection of non-technical losses in power utilities—a comprehensive systematic review. *Energies* 13(18), 4727 (2020). <https://doi.org/10.3390/en13184727>
- Badrinath Krishna, V., Iyer, R.K., Sanders, W.H.: ARIMA-based modeling and validation of consumption readings in power grids. In: *Proceedings of the International Conference on Critical Information Infrastructures Security*, pp. 199–210. Springer, Berlin (2015)
- Xia, X., Xiao, Y., Liang, W.S.A.I.: A suspicion assessment-based inspection algorithm to detect malicious users in smart grid. *IEEE Trans. Inf. Forensics Secur.* 15, 361–374 (2019). <https://doi.org/10.1109/ifs.2019.2921232>
- Viegas, J.L., et al.: Solutions for detection of non-technical losses in the electricity grid: a review. *Renew. Sustain. Energy Rev.* 80, 1256–1268 (2017). <https://doi.org/10.1016/j.rser.2017.05.193>
- Jokar, P., Arianpoo, N., Leung, V.C.: Electricity theft detection in AMI using customers' consumption patterns. *IEEE Trans. Smart Grid* 7(1), 216–226 (2015). <https://doi.org/10.1109/tsg.2015.2425222>
- Guo, Y., Ten, C.W., Jirutitijaroen, P.: Online data validation for distribution operations against cyber tampering. *IEEE Trans. Power Syst.* 29(2), 550–560 (2013). <https://doi.org/10.1109/tpwrs.2013.2282931>
- Messinis, G.M., Hatzigiorgiou, N.D.: Review of non-technical loss detection methods. *Elec. Power Syst. Res.* 158, 250–266 (2018). <https://doi.org/10.1016/j.epsr.2018.01.005>
- Ferreira, T.S.D., Trindade, F.C., Vieira, J.C.: Load flow-based method for nontechnical electrical loss detection and location in distribution systems using smart meters. *IEEE Trans. Power Syst.* 35(5), 3671–3681 (2020). <https://doi.org/10.1109/tpwrs.2020.2981826>
- Tariq, M., Poor, H.V.: Electricity theft detection and localization in grid-tied microgrids. *IEEE Trans. Smart Grid* 9, 1920–1929 (2016). <https://doi.org/10.1109/tsg.2016.2602660>
- Chen, L., Xu, X., Wang, C.: Research on anti-electricity stealing method based on state estimation. In: *Proceedings of the 2011 IEEE Power Engineering and Automation Conference*, Wuhan, China, vol. 2, pp. 413–416 (2011)
- McLaughlin, S., et al.: A multi-sensor energy theft detection framework for advanced metering infrastructures. *IEEE J. Sel. Area. Commun.* 31(7), 1319–1330 (2013). <https://doi.org/10.1109/jsac.2013.130714>
- Xiao, Z., Xiao, Y., Du, D.H.C.: Exploring malicious meter inspection in neighborhood area smart grids. *IEEE Trans. Smart Grid* 4(1), 214–226 (2012). <https://doi.org/10.1109/tsg.2012.2229397>
- Angelos, E.W.S., et al.: Detection and identification of abnormalities in customer consumptions in power distribution systems. *IEEE Trans. Power Deliv.* 26(4), 2436–2442 (2011). <https://doi.org/10.1109/tpwr.2011.2161621>
- Zheng, K., et al.: A novel combined data-driven approach for electricity theft detection. *IEEE Trans. Ind. Inf.* 15(3), 1809–1819 (2018). <https://doi.org/10.1109/tii.2018.2873814>
- Ramos, C.C.O., et al.: Identification and feature selection of non-technical losses for industrial consumers using the software weka. In: *Proceedings of the 2012 10th IEEE/IAS International Conference on Industry Applications*, pp. 1–6. Fortaleza, Brazil (2012)
- Khan, Z.A., et al.: Electricity theft detection using supervised learning techniques on smart meter data. *Sustainability* 12(19), 8023 (2020). <https://doi.org/10.3390/su12198023>
- Depuru, S.S.S.R., Wang, L., Devabhaktuni, V.: Support vector machine based data classification for detection of electricity theft. In: *Proceedings of the 2011 IEEE/PES Power Systems Conference and Exposition*, Phoenix, AZ, USA, pp. 1–8 (2011)
- Nagi, J., et al.: Non-technical loss analysis for detection of electricity theft using support vector machines. In: *Proceedings of the 2008 IEEE 2nd International Power and Energy Conference*, Johor Bahru, Malaysia, pp. 907–912 (2008)
- Nizar, A., Dong, Z., Wang, Y.: Power utility nontechnical loss analysis with extreme learning machine method. *IEEE Trans. Power Syst.* 23(3), 946–955 (2008). <https://doi.org/10.1109/tpwrs.2008.926431>
- Costa, B.C., et al.: Fraud detection in electric power distribution networks using an ann-based knowledge-discovery process. *Int. J. Artif. Intell. Appl.* 4(6), 17–23 (2013). <https://doi.org/10.5121/ijai.2013.4602>
- Hussain, S., et al.: A novel feature engineered-CatBoost-based supervised machine learning framework for electricity theft detection. *Energy Rep.* 7, 4425–4436 (2021). <https://doi.org/10.1016/j.egy.2021.07.008>
- Hussain, S., et al.: A novel feature-engineered-NGBoost machine-learning framework for fraud detection in electric power consumption data. *Sensors* 21(24), 8423 (2021). <https://doi.org/10.3390/s21248423>
- Khan, I.U., et al.: Big data analytics for electricity theft detection in smart grids. In: *Proceedings of the 2021 IEEE Madrid PowerTech*, Madrid, Spain, pp. 1–6 (2021)
- Bohani, F.A., et al.: Comprehensive analysis of supervised learning techniques for electricity theft detection. *J. Electr. Comput. Eng.* 2021, 9136206 (2021)
- Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv* (2014). [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)

31. Generalized linear model (GLM). <https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-397%20science/glm.html?highlight=glm>. Accessed: Jul-25-2023.398
32. Gradient boosting machine (GBM). <https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-399%20science/gbm.html?highlight=gbm>. Accessed: Jul-25-2023
33. Click, C., et al.: Gradient boosting machine with h2o. H2O (2017). ai
34. Alarfaj, F.K., et al.: Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access* 10, 39700–39715 (2022). <https://doi.org/10.1109/access.2022.3166891>
35. Yao, D., et al.: Energy theft detection with energy privacy preservation in the smart grid. *IEEE Internet Things J.* 6(5), 7659–7669 (2019). <https://doi.org/10.1109/jiot.2019.2903312>
36. Muniz, C., et al.: Irregularity detection on low tension electric installations by neural network ensembles. In: *Proceedings of the 2009 International Joint Conference on Neural Networks*, Atlanta, GA, USA, pp. 2176–2182 (2009)
37. Depuru, S.S.S.R., Wang, L., Devabhaktuni, V.: Enhanced encoding technique for identifying abnormal energy usage pattern. In: *Proceedings of the 2012 North American Power Symposium (NAPS)*, Champaign, IL, USA, pp. 1–6 (2012)
38. Aiello, S., et al.: *Machine Learning with R and H2O*. H2O booklet (2016).550
39. Badawi, S.A., et al.: A novel time-series transformation and machine-learning-based method for NTL fraud detection in utility companies. *Mathematics* 10(11), 1878–2022 (2022). <https://doi.org/10.3390/math10111878>
40. Kulkarni, Y., et al.: EnsembleNTLDetect: an intelligent framework for electricity theft detection in smart grid. In: *2021 International Conference on Data Mining Workshops (ICDMW)*, pp. 527–536. *IEEE Access* (2021)
41. Mujeeb, S., et al.: An efficient electricity theft detection scheme with additive 404 communication layer. In: *Proceedings of the ICC 2020—2020 IEEE International Conference on Communications*, Dublin, Ireland, vol. 405, pp. 1–6 (2020)
42. Ullah, A., et al.: CNN and GRU based deep neural network for electricity theft detection to secure smart grid. In: *Proceedings of the 2020 International Wireless Communications and Mobile Computing (IWCMC)*, Limassol, Cyprus, pp. 1598–1602 (2020)

How to cite this article: Badawi, S.A., et al.: A novel two-stage method to detect non-technical losses in smart grids. *IET Smart Cities*. 1–16 (2024). <https://doi.org/10.1049/smc2.12078>