

ORIGINAL RESEARCH

Open access

Advanced Fraud Detection in Smart Grids: A Two-Phase Method with Machine Learning

Ahmed Youssef^{1*}, Khaled Hassan¹, Mahmoud Elamin²

Abstract

A wide range of strategies has been proposed to identify and prevent non-technical electricity losses resulting from fraudulent activities. Among these, machine learning algorithms and data-driven methods have become increasingly prominent compared to conventional approaches, owing to their superior performance, leading to growing adoption in recent years. This paper introduces a novel two-step method for detecting fraudulent Non-technical losses (NTLs) in smart grids. The initial step is to transform the time-series data by incorporating additional features extracted from the publicly available State Grid Corporation of China (SGCC) dataset. These features are derived following the detection of sudden changes in electricity consumption patterns, using the sum of finite differences, the Auto-Regressive Integrated Moving Average model, and the Holt-Winters model. Subsequently, five classification models are applied to train and evaluate a fraud detection system using the SGCC dataset. The performance evaluation shows that the Gradient Boosting Machine delivers the best results among the five models. This two-step procedure enables classification models to outperform previously reported top-performing techniques in terms of accuracy, F1-score, and other key metrics for non-technical loss detection.

Keywords Smart cities, Data analytics and machine learning, Data structures, Power meters, Power system security, Smart power grids

*Correspondence:

Ahmed Youssef
ayoussef@uofk.edu

¹ Department of Intelligent Power Systems, Faculty of Engineering, University of Khartoum, Khartoum, Sudan

² Department of Energy Systems Optimization, Faculty of Engineering, Sudan University of Science and Technology, Khartoum, Sudan

Introduction

Smart cities are modern urban environments that leverage advanced technologies and data-driven strategies to promote sustainability, optimize resource allocation, and enhance residents' quality of life. A central element of this transformation is the smart grid, an advanced electricity distribution network that forms the backbone of a city's energy infrastructure. By integrating renewable energy sources, improving operational efficiency, and increasing environmental responsibility, the smart grid fundamentally reshapes energy management practices. Smart meters play a crucial role in this framework by collecting real-time data, which empowers consumers to monitor and manage their electricity consumption effectively. Moreover, these meters enable utility providers to quickly detect and address issues such as electricity theft, outages, and other forms of energy loss, thereby maintaining a reliable and secure power supply.

Electrical power losses are generally classified into two categories: technical losses and non-technical losses (NTL) [1]. Technical losses occur during the transmission of energy from generation plants to final consumers and manifest as conversions into forms such as heat in power lines and transformers [2]. These losses can be reduced by modifying or

redesigning components of the grid [3]. Non-technical losses (NTL), in contrast, refer to energy delivered but not billed, typically due to theft or fraudulent activity. In most countries, the main sources of NTLs are tampering with smart meters and unauthorized alterations to their recorded measurements [4]. Consequently, utility companies invest considerable resources and effort into identifying fraudsters and safeguarding power systems against unauthorized interference.

NTLs can be effectively reduced by applying various techniques that produce actionable insights for detecting malicious users. It is also widely considered more practical to prioritize NTL minimization before tackling technical losses [5]. Researchers typically classify NTLs into two main types: 1) direct tampering with meter readings to under-report usage or complete bypassing of the meter (often under the guise of official technical testing), and 2) NTLs arising from internal manipulations, which involve corrupt or dishonest actions by employees of the power utility or company [4]. Correspondingly, NTL detection methods fall into two broad categories: hardware-based approaches and non-hardware (alternative) approaches. Hardware-based techniques rely on specialized intelligent devices installed at targeted points in the electrical network to identify fraud. Non-hardware alternatives include network-based, data-based, and hybrid methods, as documented in the current literature. These methods have developed in parallel with advances in digital technology, facilitating the collection of extensive, varied data on user consumption behavior.

This work investigates data-driven approaches employing machine learning (ML) algorithms, a rapidly growing field in non-technical loss (NTL) detection research, largely because these methods offer clear advantages over traditional hardware-based techniques [6]. The supervised ML algorithms evaluated in this study include Gradient Boosting Machine (GBM), Generalized Linear Model (GLM), Deep Learning (DL), and Naive Bayes (NB). Experiments were performed using a two-step procedure applied to the dataset from the State Grid Corporation of China (SGCC). In the first step, sudden and irregular deviations from a consumer's normal consumption behavior are identified by statistically analyzing variations in usage patterns in relation to the moving midpoint of a sliding time window. The well-established time-series forecasting method, the Auto-Regressive Integrated Moving Average (ARIMA) model, is used to predict future consumption based on historical usage data [7]. Drawing on the outcomes of this initial step, a new set of features is constructed for the ML algorithms. In the second step, the selected ML models are applied to classify whether the consumer's behavior is fraudulent. The primary contributions and objectives of this research are outlined below:

1. Presentation of an innovative data transformation approach for smart meter readings that combines the sum of finite differences to capture abrupt spikes in electricity consumption, along with ARIMA and Holt-Winters models, to derive statistical features for NTL detection. This proposed transformation improves the fraud-detection performance of the evaluated ML algorithms.
2. A key strength of the proposed method lies in its incorporation of finite differences as features within the transformed dataset, together with the demonstrated effectiveness of the GBM algorithm in identifying NTL fraud.
3. Assessment of the proposed method and direct comparison with existing literature approaches that also utilize the SGCC dataset. Experimental results demonstrate that the suggested technique surpasses current state-of-the-art methods across accuracy, precision, recall, and F1-score metrics.

Literature Review

Methods for detecting and identifying non-technical losses (NTLs) caused by fraudulent behavior toward utility customers are broadly classified into two groups: hardware-based and non-hardware-based techniques. Utility companies can detect potential fraudulent activity [8, 9] by installing dedicated meters, which serve as the core component of hardware-oriented detection systems. Non-hardware alternatives to these approaches fall into three main categories: 1) data-based methods, 2) network-based techniques, and 3) hybrid methods combining elements of the first two [6, 10, 11]. Network-based techniques gather information from observer meters, smart meters, and strategically placed sensors throughout the grid. Several comprehensive review papers and related studies [6, 12] have thoroughly examined and cataloged these approaches. The literature further divides network-based methods into four distinct categories:

1. Load flow approach [13, 14]: Customer electricity consumption is monitored by placing an observer meter at a selected point in the grid. The measurements obtained are then compared with data from individual customer smart meters to reveal discrepancies indicative of fraud.
2. Condition prediction methodology [9, 15]: The grid's state is monitored using data collected from smart meters.
3. Sensing network methodology [16, 17]: Dedicated sensors are deployed at targeted locations within the grid to track consumption patterns.

Data-driven methods leverage consumption information collected from smart meters over a specified observation period and apply various machine learning (ML) algorithms to derive personalized behavioral patterns. Machine learning approaches provide multiple benefits, including improved accuracy, greater efficiency, reduced processing time, and lower human resources requirements [6]. ML-based NTL detection methods are generally divided into supervised and unsupervised categories [18, 19]. Unsupervised techniques for NTL detection rely on unlabeled customer data, whereas supervised methods utilize labeled data marked as “Fraud” or “Non-fraud”. Among the most widely studied supervised techniques in existing literature are Artificial Neural Networks (ANN), Support Vector Machines (SVM), Convolutional Neural Networks (CNN), Optimum Path Forest, Decision Trees (DT), and several ensemble learning algorithms such as AdaBoost, XGBoost, and Random Forest (RF).

SVMs have frequently served as the main approach for NTL detection [14, 20]. They perform well on small datasets but show reduced effectiveness when applied to large, imbalanced datasets [21]. In some studies, SVM achieved accuracies ranging from 86% to 98% [22, 23]. The Multilayer Perceptron has also emerged as a popular method for NTL detection [2]. It was employed by Nizar *et al.* [24] and Costa *et al.* [25], who used features derived from statistical and spectral analyses of time-series data, yielding accuracies of 54.61%–87.11%.

Hussain *et al.* [26] introduced CatBoost, a novel supervised ML technique for identifying NTLs. The authors applied it to the SGCC dataset—an authentic, publicly released dataset from the State Grid Corporation of China containing electricity consumption records of 42,372 customers across 1035 days. Additionally, the Feature Extraction and Scalable Hypothesis algorithm was utilized to extract and select the most relevant temporal, statistical, and spectral features. This approach achieved 93.4% accuracy and 95% precision. When the same analysis was conducted using RF, the accuracy dropped to 87%, compared with other supervised methods such as extreme gradient boosting, DT classification, light gradient boosting, and AdaBoost, which produced accuracies ranging from 81% to 91%. In the initial stage, the authors applied data class balancing techniques to improve result reliability; however, these steps had only a minor effect on accuracy, recall, and precision.

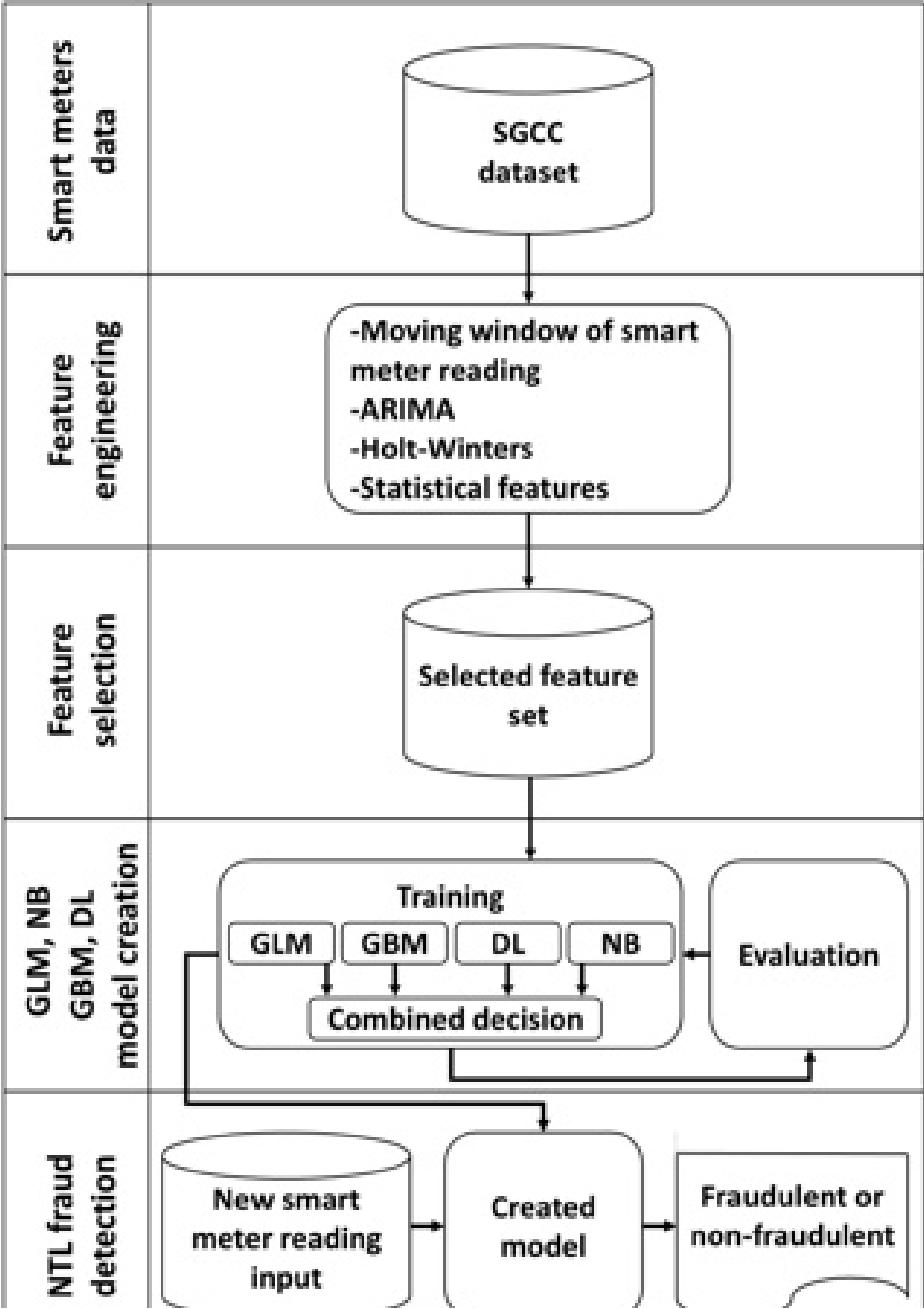
Hussain *et al.* [27] further proposed an advanced machine learning framework, NGBoost, for NTL detection. They employed the Time Series Feature Extraction Library and the whale-optimization algorithm to generate statistical, temporal, and spectral features. In terms of performance, NGBoost successfully classified consumers into “Healthy” and “Theft” categories and outperformed RF, CatBoost, AdaBoost, DT, and gradient boosting.

Khan *et al.* [28] used a Bayesian SVM to classify consumers in the SGCC dataset as either “truthful” or “deceptive”. By applying Bayesian optimization, their classification model achieved 94.1% accuracy, surpassing RF, Logistic Regression (LR), and standard SVM, and improving hyper-parameter optimization and overall learning accuracy. Bohani *et al.* [29] conducted a comparative evaluation of several supervised ML algorithms—including DT, ANN, Deep Neural Networks (DNN), and AdaBoost—using the SGCC dataset. Among them, the DNN algorithm achieved the highest accuracy, 93.04%. However, the results were considered less reliable owing to relatively low recall and F1 scores. Khan *et al.* [21] proposed and tested a novel method, Extra Gradient Boosting (FA-XGBoost), on the SGCC dataset. The Visual Geometry Group tool was used for feature extraction, although the additional computational burden was not a primary focus [30]. For comparison, established techniques such as SVM, CNN, and LR were also assessed. The proposed

system achieved an impressive 95% accuracy. For classification purposes, the authors' framework incorporates a supervised ML algorithm known as Distributed Random Forest (DRF).

Materials and Methods

The proposed method (**Figure 1**) consists of a two-stage feature extraction procedure. The initial stage focuses on data analysis, preprocessing, and feature derivation from the SGCC dataset. The core assumption underlying the data analysis is that abnormal spikes in a consumer's average power consumption are likely indicative of fraudulent behavior. For instance, **Figure 2** demonstrates how smart meter readings can abruptly shift to significantly lower consumption levels following a specific pattern. Likewise, when consumption differences are calculated in relation to a neighboring meter, a sharp increase in the adjacent smart meter's readings becomes apparent (**Figure 3**). During this stage, the dataset is preprocessed to generate a comprehensive set of features for each smart meter reading. These features are computed using a 2-month sliding window that includes 1 month of prior (lag) data and 1 month of subsequent (lead) data for each reading. The first group of features is formed by calculating finite-difference differences across these readings. Subsequently, patterns in the moving averages of the smart meter readings are analyzed using optimized moving-average techniques that incorporate seasonality evaluation, the Holt-Winters method, ARIMA, and trend analysis. This process yields the second group of features.



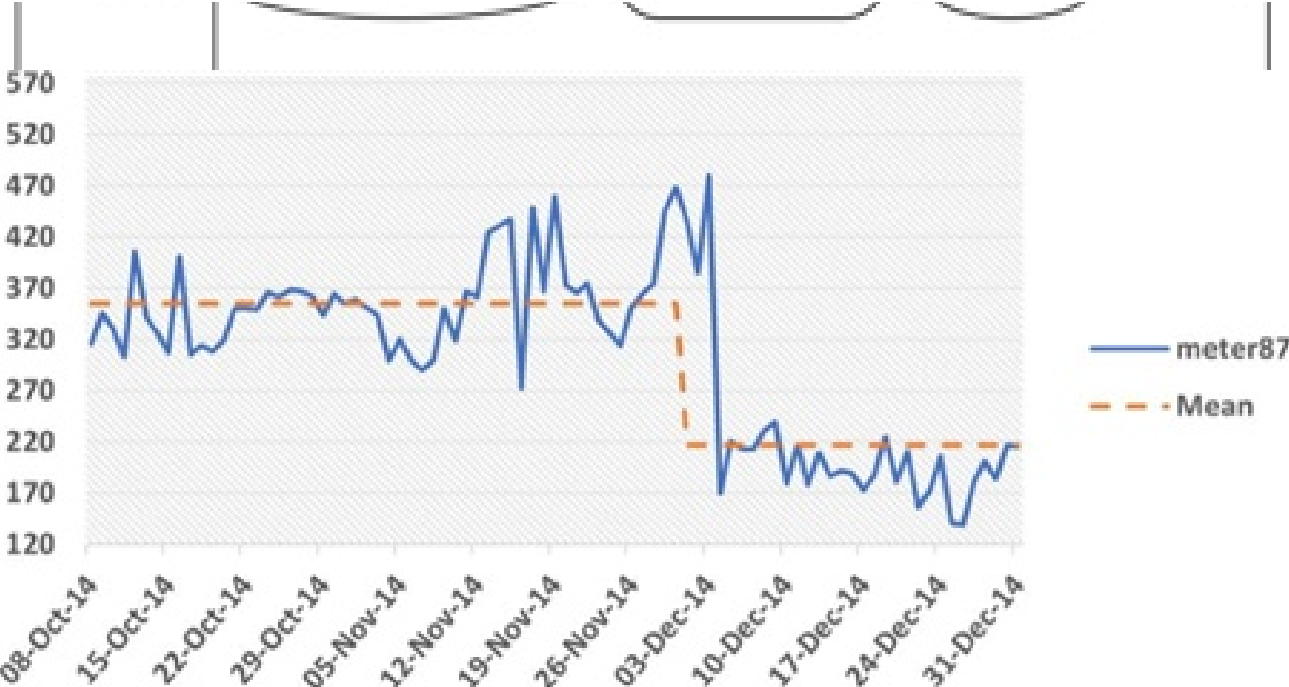


Figure 2. The attacker’s meter exhibits an abrupt decline in median consumption, reflecting a substantial reduction in smart meter readings. Blue represents the actual consumption recorded by smart meters. At the point of fraud, an orange spike appears in the average of the lagging meter readings and the average of the leading meter readings.

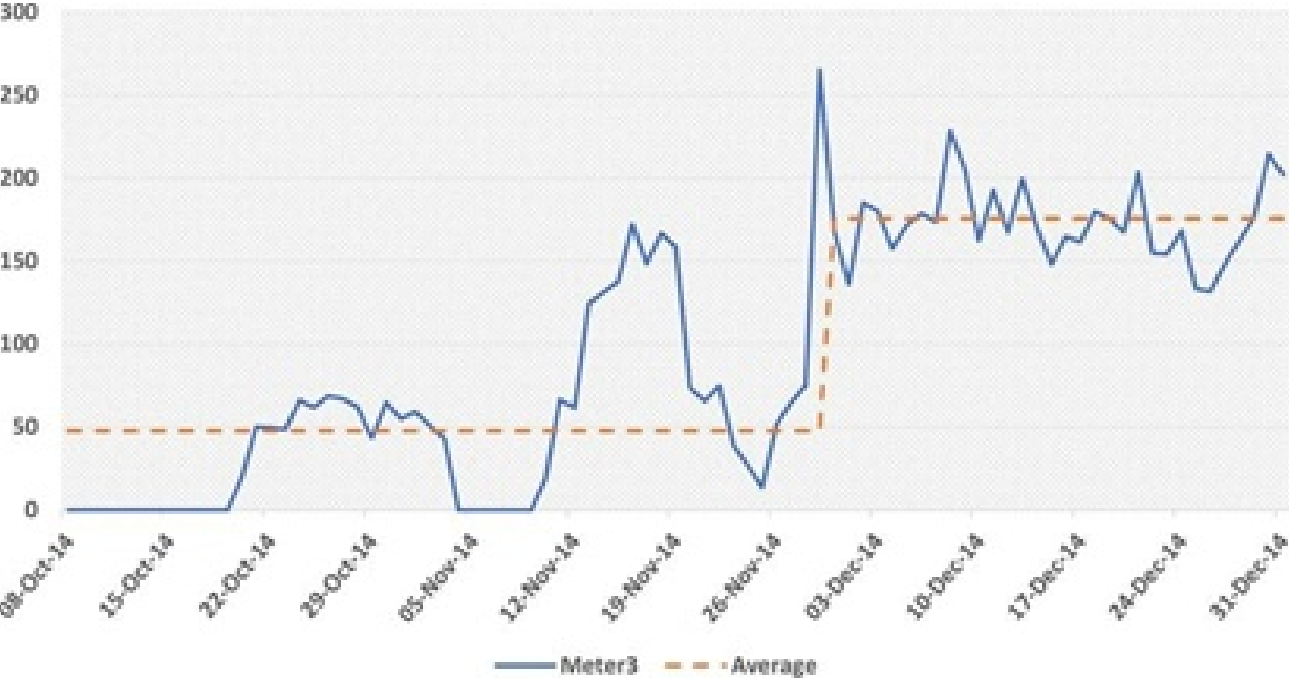


Figure 3. Detection of a sudden increase in the average consumption of the tampered smart meter. The blue line shows the usage recorded by the smart meter, while the orange line indicates the average of the lagging and leading smart meter readings.

Feature engineering

Initially, a fresh set of features was created by leveraging readings from a sliding window of leading and lagging values derived from a normalized version of the SGCC dataset, along with the corresponding statistical measures. Normalization was performed using min-max scaling (Eq. 1), mapping values to the interval [0, 1]. The construction of these new feature sets and their statistical properties are described in the sections that follow.

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

Where z denotes the original feature value, $Norm$ represents the normalized value, $\max(z)$ is the maximum value of z , and $\min(z)$ is the minimum value of z .

The identification of abrupt surges in smart meter consumption

To detect potential fraud, the suggested method identifies sudden behavioral shifts by constructing a dataset that includes attributes for each of the 61 readings within a sliding window. As depicted in Figures 2 and 3, this dataset comprises engineered features that capture the temporal context around each reading. These features encompass statistics such as the pre-point and post-point means, pre-point and post-point medians, pre-point and post-point variances calculated over a span of 32 readings before and after, as well as 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 represents half the window size, Lag_j denotes the j th lag reading, and $Lead_j$ denotes the j th lead reading.

Retrieval of characteristics from smart meter data

Following the detection of unexpected jumps, an additional set of statistical features is added to the time-series dataset. Table 1 provides a summary and overview of these statistical features. Once the smart meter data is transformed into a stationary series (with constant mean and variance), various characteristics can be extracted, including those derived from ARIMA, Holt-Winters, trend, and seasonality components. The ARIMA model is particularly well-suited for improving forecasting precision, as it uses historical time-series data to predict future patterns.

Table 1. A set of specially designed attributes intended for the detection of fraudulent activities.

Generated Feature	Definition
Smart meter value moving window	Time series readings from the smart meter (32 preceding values + current value + 32 following values)
General statistical features	Sum, mean, median, variance, and other basic statistics
$\delta 1$ to $\delta 32$ (seasonality, randomness, and trend)	$\delta(i) = Lead(i) - Lag(i)$, derived from seasonal-trend decomposition results
Sum of finite differences	Sum of $\Delta(i)$ values calculated between the current reading and its neighboring time series points

Holt-Winters	Exponential smoothing method capable of ignoring outliers or irrelevant readings
ARIMA	Autoregressive Integrated Moving Average model
Label	Binary indicator: 1 = fraud, 0 = non-fraud

Feature selection

The Boruta algorithm is widely recognized and frequently applied for feature selection. It introduces random shadow variables (permutations of the original features) to identify which actual features consistently rank higher than their randomized counterparts. Gradient Boosting Machine exhibits a certain degree of robustness against the inclusion of irrelevant random features, owing to its iterative nature: it repeatedly evaluates and re-scores data points, particularly when row subsampling is employed. Subsequent iterations can correct earlier suboptimal splitting decisions. By monitoring the performance of these random shadow variables, the algorithm helps avoid erroneous splits on genuine features.

Classifications

As previously outlined, this study examines four distinct classification approaches—namely, GLM, RF, GBM, and DL. These machine learning algorithms are applied to the SGCC dataset to classify smart meter consumption records as either fraudulent or legitimate. The obtained results are subsequently benchmarked against findings from comparable methods documented in the existing literature.

Generalized linear model

The Generalized Linear Model (GLM) is a well-established and widely adopted machine learning technique for detecting fraudulent activity in smart meters. It serves as a versatile and powerful statistical framework capable of capturing intricate relationships between predictor variables and the probability of fraudulent behavior. GLM offers extensive configurability through various hyperparameters, including the choice of distribution family (Gaussian, Poisson, Binomial), the selection of the link function (identity, log, logit, inverse), and regularization parameters (alpha and lambda), as well as strategies for handling missing data [31]. By leveraging the flexibility and modeling power of GLMs and carefully tuning their hyperparameters, the present study aims to achieve efficient and reliable identification of deceptive patterns in smart electricity meter readings, thereby supporting accurate analysis and preserving the reliability of energy consumption records.

Gradient boosting machine

Gradient Boosting Machine (GBM) is a highly effective and popular machine learning algorithm known for handling complex datasets and delivering superior predictive performance. It constructs an ensemble by sequentially combining weak learners—most commonly decision trees [32]. Each new model in the sequence is trained to correct the residuals (errors) produced by the previous models, progressively improving the ensemble's overall predictive power. Thanks to its proficiency at capturing non-linear patterns and subtle interactions in data, GBM has proven successful across numerous application areas, including fraud detection.

GBM constitutes one of the primary algorithms evaluated in this research. Its performance is optimized by fine-tuning key hyperparameters, including the learning rate, the total number of trees, and the maximum depth of individual trees. The learning rate controls the contribution of each tree to the final prediction, while the number of trees and maximum tree depth determine the model's capacity and resilience [33]. By systematically evaluating GBM and its hyperparameter

configurations, this work aims to assess its effectiveness in accurately detecting fraudulent electricity consumption patterns, thereby advancing the development of robust fraud-detection mechanisms for smart grid environments.

The GBM model was evaluated in two separate configurations differing in the number of trees. The first configuration used the default setting of 50 trees. The second configuration increased this to 500 trees (Table 2). To mitigate the risk of overfitting, early stopping criteria were implemented. Three specific stopping-related hyperparameters were defined: the stopping metric was set to Area Under the Curve (AUC), the score tree interval was established at every five trees to evaluate model performance, and the stopping rounds parameter was configured to halt training after three consecutive rounds without improvement.

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

Deep learning

Deep Learning (DL), a specialized branch of machine learning, focuses on training Artificial Neural Networks (ANNs) to automatically discover hierarchical feature representations from raw data [34-37]. This capability allows DL models to identify complex patterns and dependencies within large-scale datasets. When applied to fraud detection in smart electricity meters, DL provides several key benefits. It effectively processes high-dimensional, non-linear data, making it well-suited for modeling diverse energy consumption behaviors and the features extracted from them. Additionally, DL models exhibit strong generalization performance, enabling them to adapt to varying types of fraudulent patterns and to detect subtle anomalies that traditional rule-based or purely statistical approaches might overlook.

Given its frequent appearance in the literature as a promising ML technique, Deep Learning is also evaluated in the present study. DL leverages multi-layered neural networks to extract intricate patterns and features from input data. This research explores DL's effectiveness in fraud detection by leveraging its ability to handle high-dimensional, non-linear relationships. Key hyperparameters of the neural network architecture include the total number of hidden layers and the number of nodes per layer. Furthermore, regularization-related hyperparameters—such as the dropout rate and weight decay—can be tuned to improve generalization and mitigate overfitting.

The DL model was evaluated across three separate configurations, differing primarily in the number of training epochs. In the first experiment, the default setting of 10 epochs was used, with early stopping enabled and the standard stopping criteria applied. In the second experiment, the number of epochs was increased to 20, and early stopping was deliberately disabled by setting the stopping rounds parameter to 0. For the third experiment, the same model configuration as the second run (dl fit2) was retained, but early stopping was re-enabled with explicitly defined stopping criteria; a separate validation set was also provided, as recommended when early stopping is active (i.e., when stopping rounds > 0). The hyperparameters used for dl fit2 are listed in Table 3, while those for dl fit3 are presented in Table 4. Increasing the number of epochs in a deep neural network can further enhance model performance when optimal settings are identified. Once the most effective hyperparameter values were determined, the DL model was trained and assessed using the SGCC time-series dataset.

Table 3. Deep learning (DL) hyperparameters (dl fit2).

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

Table 4. Deep learning (DL) hyperparameters (dl fit3).

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

Naive bayes

Naive Bayes is a probabilistic classification algorithm often employed as an alternative to decision tree-based methods. It applies Bayes' theorem under the strong assumption that predictor variables (features) are conditionally independent given the class label. More specifically, the algorithm presumes that all features are independent of one another once the target class is known. For numeric predictors, it further assumes that their values follow a Gaussian (normal) distribution, with means and standard deviations estimated from the training data. During model training, any row in the training dataset containing at least one missing value is completely discarded. In prediction, if the test data contains missing values for certain predictors, those features are excluded from the probability computations.

The Naive Bayes (NB) classifier is one of the machine learning algorithms examined in this study. Like the other methods, its performance can be influenced by adjusting configurable hyperparameters, leading to varying outcomes across different settings. One notable tunable hyperparameter is the Laplace smoothing parameter. Due to the simplifying independence assumption, the NB algorithm can struggle in practical scenarios when default settings are used, particularly when zero probabilities arise. Laplace smoothing is a regularization technique that addresses the problem of zero-probability counts by adding a small constant to frequency counts.

The NB algorithm was evaluated in two distinct experiments, differing primarily in the application of Laplace smoothing. By default, the NB model applies no smoothing. In the first experiment, a baseline NB model was trained using default parameters. In the second experiment, Laplace smoothing was introduced by setting the Laplace parameter to 6. Once the hyperparameters were configured and training completed, both models were assessed using the SGCC dataset.

Experimental Evaluation

Dataset preparation

The proposed model's ability to identify electricity theft and fraud based on smart meter readings was validated using the SGCC dataset [2]. The SGCC dataset contains smart meter electricity consumption records from more than 32,000 customers collected over 1035 days. A two-stage classification process was applied to this dataset. For the experiments, a subset consisting of 7000 entries was selected, including 520 instances labeled as fraudulent users and 6480 labeled as legitimate users. After transformation, the 7000 meters (520 fraudulent versus 6480 non-fraudulent), each originally comprising 1035 readings, were converted into a total of 7,245,000 feature records. Among these, 538,200 records correspond to fraudulent cases and 6,706,800 to non-fraudulent cases.

Performance measures

The effectiveness of the machine learning algorithms was assessed using the evaluation metrics listed in Table 5. In the formulas for certain metrics, FP, TP, FN, and TN denote False Positive, True Positive, False Negative, and True Negative, respectively. Additionally, AUC stands for Area Under the Curve, MSE for Mean Squared Error, RMSE for Root Mean Squared Error, and $\bar{y}$ for the mean (or expected value) of the y values.

Table 5. Metrics employed to assess the performance of the various machine learning (ML) techniques.

Performance measure	Definition	Formula
Accuracy (Acc)	The proportion of both fraudulent and legitimate cases that are correctly classified.	$Acc = \frac{(TP + TN)}{(TP + FP + FN + TN)}$
Recall or sensitivity (Sn)	Measures how effectively the model detects actual fraudulent instances (true positive rate).	$Sensitivity = \frac{TP}{(TP + FN)}$
Specificity (Sp)	Measures how effectively the model identifies non-fraudulent instances (true negative rate).	$Specificity = \frac{TN}{(TN + FP)}$
AUC	Area under the ROC curve, indicating overall model discrimination ability.	$AUC = \frac{Rank_{k, positiveClass} - \frac{M(1+M)}{2}}{M \cdot N}$
Precision (Pr)	The fraction of predicted fraud cases that are actually fraudulent.	$Pr = \frac{TP}{(TP + FP)}$
F1-score	The harmonic mean of precision and recall balances both metrics.	$F1score = 2 \times \frac{(Pr \times SP)}{((Pr + Sp))}$
MSE (Mean Squared Error)	The average of the squared differences between predicted and actual values.	$MSE = \frac{1}{n} \sum_1^n (\bar{y}_i - y_i)^2$
RMSE (Root Mean Squared Error)	The square root of the mean squared error represents error in the same units as the target variable.	$RMSE = \sqrt{\frac{\sum_1^n (\bar{y}_i - y_i)^2}{n}}$
Logloss	Logarithmic (cross-entropy) loss that penalizes confident wrong predictions.	$LogLoss = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^M x_{ij} \log(p_{ij})$

R² (Coefficient of Determination)	The proportion of variance in the dependent variable that is explained by the independent variables in a regression model.	$R^2 = 1 - \frac{\sum_1^n (\hat{y}_i - y_i)^2}{\sum_1^n (\hat{y}_i - \bar{y})^2}$
---	--	---

Results and Discussion

In this study, the engineered dataset was partitioned into separate training and validation (testing) subsets. Four machine learning algorithms were implemented in R. Data processing and transformation were performed using the ‘dplyr’ package. Statistical feature computations were performed using the ‘matrixStats’ package. Auto-Regressive Integrated Moving Average feature extraction was conducted with the ‘Forecast’ package. Model training, implementation, and visualization relied on the ‘tidyverse’, ‘H2O’ [38], and ‘ggplot2’ packages. All machine learning models were trained on a computing system equipped with a Core i7 processor, 32 GB of RAM, and a 12 GB NVIDIA GTX graphics card. Detailed performance outcomes for each algorithm are presented in the subsections below.

Generalized linear model results

The outcomes of the two experiments conducted with the GLM algorithm indicate that it consistently achieved a sensitivity (recall) of 1.0 across both runs. In the first experiment, the model achieved an accuracy of 0.999782, specificity and precision of 0.999995, an F1-score of 0.999783, and an AUC of 0.999995. In the second experiment, the accuracy reached 0.999,801, with specificity and precision both at 1.0, an F1-score of 0.999,801, and an AUC of 0.9999989. A concise summary of the GLM experimental results is provided in **Tables 6 and 7**. Visual representations of these results are illustrated in **Figure 4**.

Table 6. Generalized 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. Generalized linear model (GLM) 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

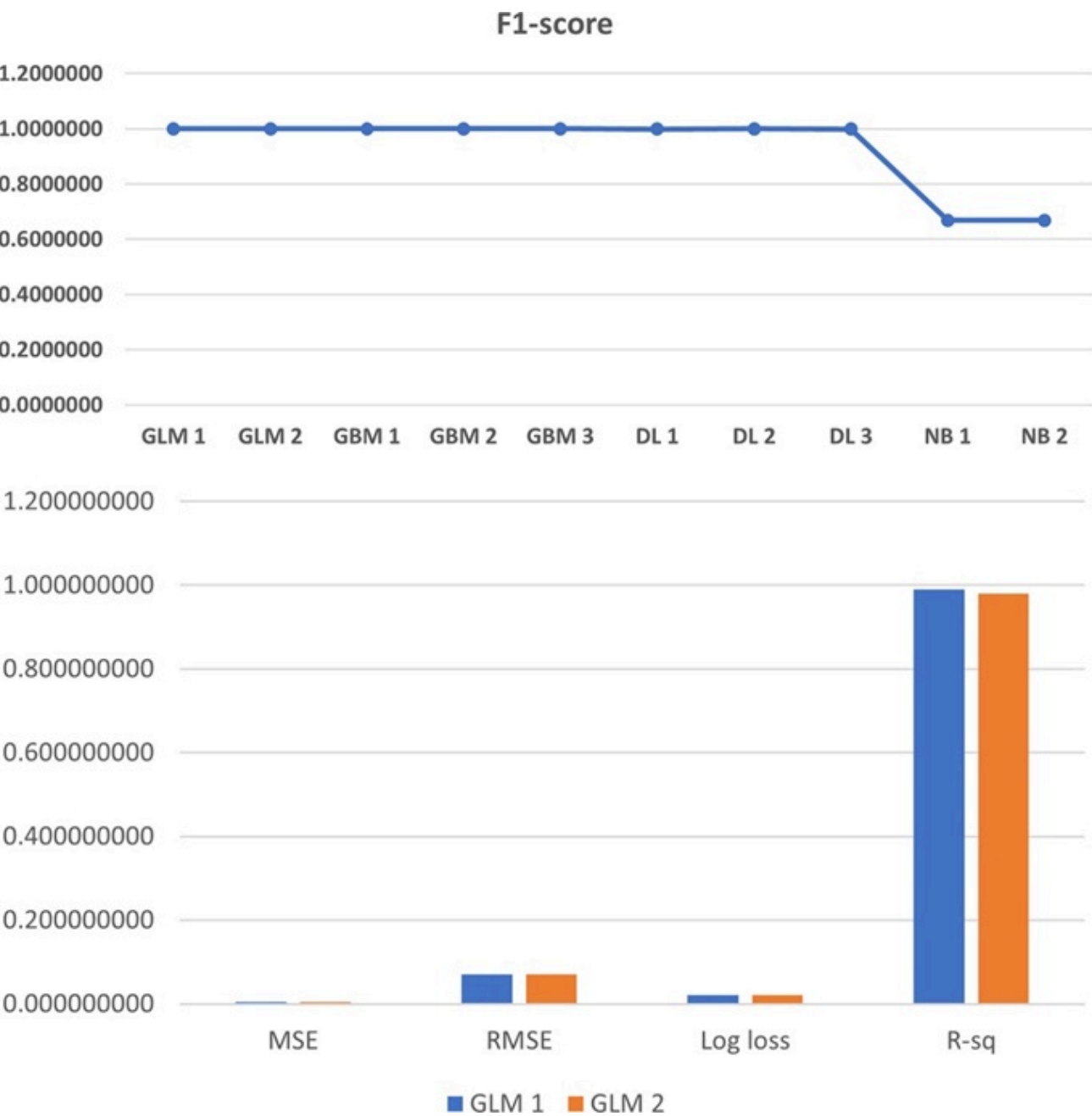


Figure 4. Results of the generalized linear model (GLM) experiments. The upper sub-Figure displays the performance metrics, while the lower sub-Figure illustrates the model losses.

Gradient boosting machine results

Both trials involving the gradient boosting machine (GBM) delivered remarkably strong outcomes. Across the two runs, the GBM classifiers performed excellently, achieving perfect scores of 1.0 on all performance indicators (accuracy, sensitivity, specificity, precision, F1-score, and AUC). Although slight differences appeared in the loss values and R² Figures, these variations remained well aligned with the models' consistently superior effectiveness. For the initial experiment, the recorded MSE was 0.000,010569, RMSE was 0.003251006, Logloss was 0.00,325624, and R² was 0.9999577. In the follow-up experiment, every loss metric dropped to 0.0, and R² climbed to 1.0, clearly outperforming the first run and confirming a more robust model overall. Model loss details appear in [Table 8](#), with corresponding visual summaries presented in [Figure 5](#).

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

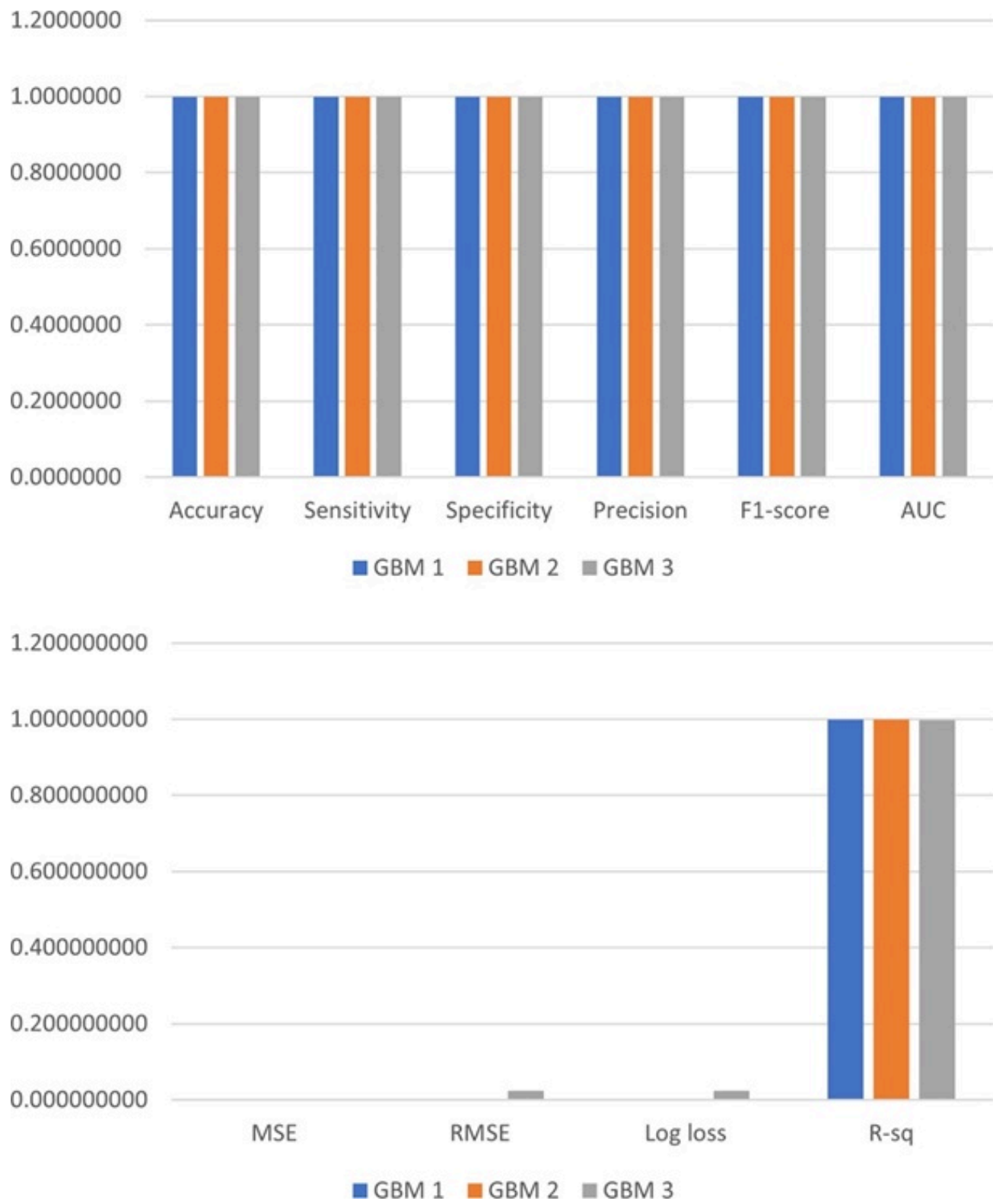


Figure 5. Outcomes from the gradient boosting machine (GBM) trials. The upper panel displays key performance indicators; the lower panel presents the associated model losses.

Deep learning results

The Deep Learning (DL) approach was subjected to three distinct experimental runs. In the opening trial, the network demonstrated strong performance, achieving an accuracy of 0.999,283, a perfect sensitivity of 1.0, specificities and precision both at 0.999,937, an F1-score of 0.999,286, and an AUC of 0.9998998837. Corresponding error metrics were MSE = 0.000,685,124, RMSE = 0.02617488, and logloss = 0.005,926,242.

The second trial produced results that matched or slightly surpassed those of the first in multiple dimensions. Accuracy improved to 0.999,844, sensitivity held steady at 1.0, specificity reached 0.999,912, precision stood at 0.999,913, the F1-score climbed to 0.999,845, and AUC reached 0.9999303. Error measures were noticeably reduced: MSE = 0.000154854, RMSE = 0.01244404, and logloss = 0.002856985.

Performance in the third trial, while still very high overall, fell somewhat short of the earlier two. Accuracy was recorded at 0.999395; sensitivity remained perfect at 1.0; specificity and precision both settled at 0.999486; the F1-score was 0.999398; and AUC was 0.9997067. Loss Figures for this configuration were MSE = 0.000,952,688, RMSE = 0.03086564, and logloss = 0.008,014,885. Note that R^2 was not calculated for any of the DL configurations. Comprehensive performance and loss summaries for all three experiments are compiled in [Tables 9 and 10](#). [Figure 6](#) provides a graphical overview of these findings.

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

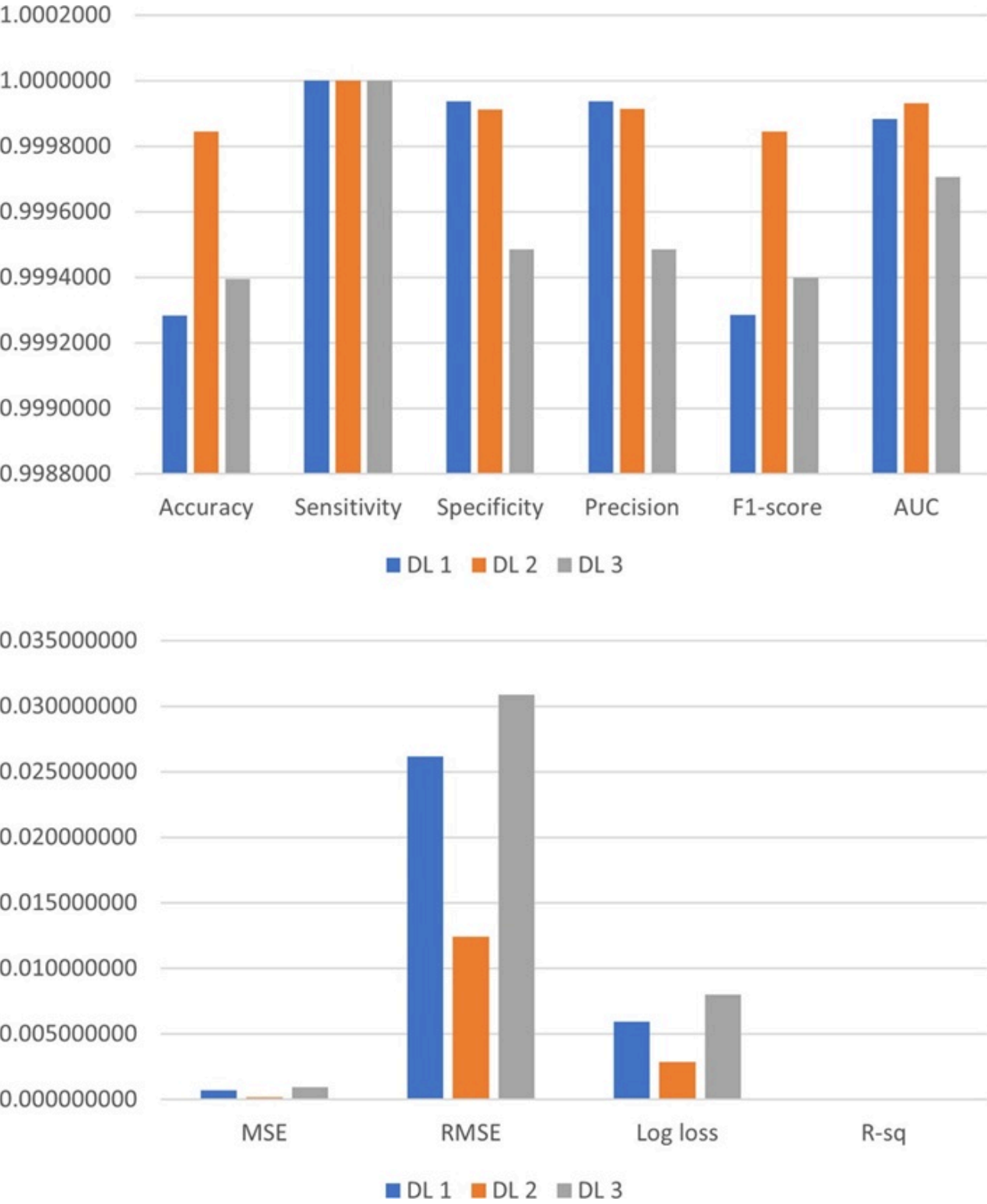


Figure 6. Visual summary of deep learning (DL) experiments. Performance indicators are in the top portion; model loss values are in the bottom portion.

Naive bayes results

The Naive Bayes (NB) classifier was the final algorithm evaluated. Remarkably, the two NB experiments produced virtually identical outcomes, suggesting that adjustments to the hyperparameters had no meaningful effect on classification performance. In each case, the model achieved an accuracy of 0.52. Sensitivity registered a perfect 1.0, specificity reached 0.986,105, and precision was 0.796,137. This translated into an F1-score of 0.668,538 and an AUC of 0.5219506. Error metrics included MSE = 0.4811734, RMSE = 0.6936667, and a substantially elevated logloss of 16.55273. Performance indicators are tabulated in [Table 11](#); loss metrics appear in [Table 12](#). [Figure 7](#) offers a graphical depiction of the complete set of results.

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

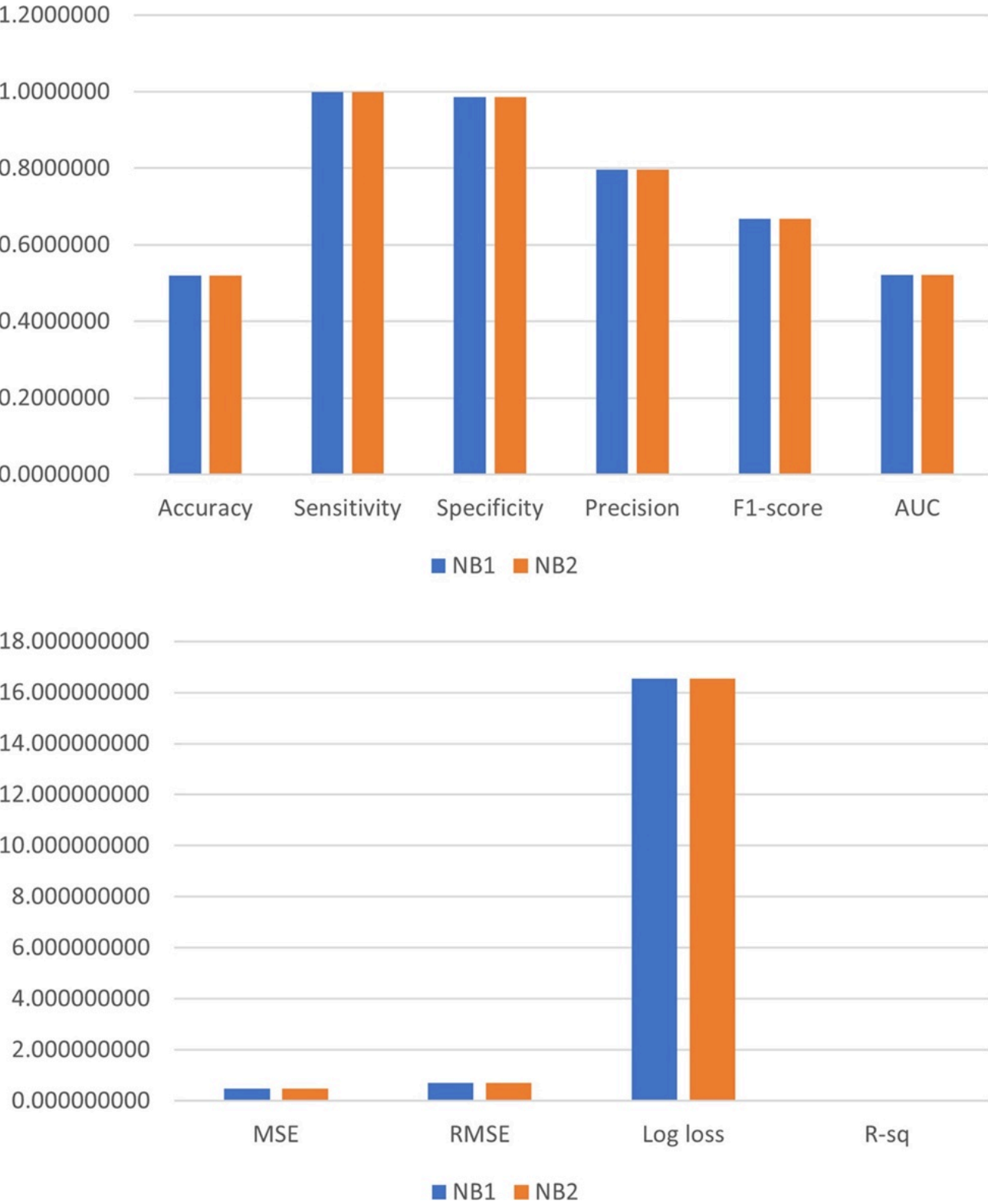


Figure 7. Graphical representation of Naive Bayes (NB) experiment outcomes. Performance metrics are displayed in the upper panel, and model losses are displayed in the lower panel.

The primary aim of the experiments was to identify the most effective machine learning algorithm for the SGCC dataset. These findings closely align with the anticipated results and support the broader objective of developing a robust, high-

performance solution to detect non-technical loss (NTL) fraud. Four machine learning classifiers were assessed: GLM, GBM, DL, and NB. As described earlier, the processed dataset was split into separate training and evaluation portions. All experiments were implemented in the R programming environment, utilizing the previously specified libraries and configurations. A side-by-side visual comparison of the performance and loss metrics across all tested models is presented in **Figures 8 and 9**. The analysis clearly indicates that the Gradient Boosting Machine (GBM) achieved the strongest overall performance on the SGCC dataset. Notably, the second GBM experiment delivered the highest values across all performance indicators while simultaneously recording the lowest loss metrics. Based on these outcomes, the GBM model is recommended as a highly accurate and reliable classifier for identifying electricity theft and non-technical losses.

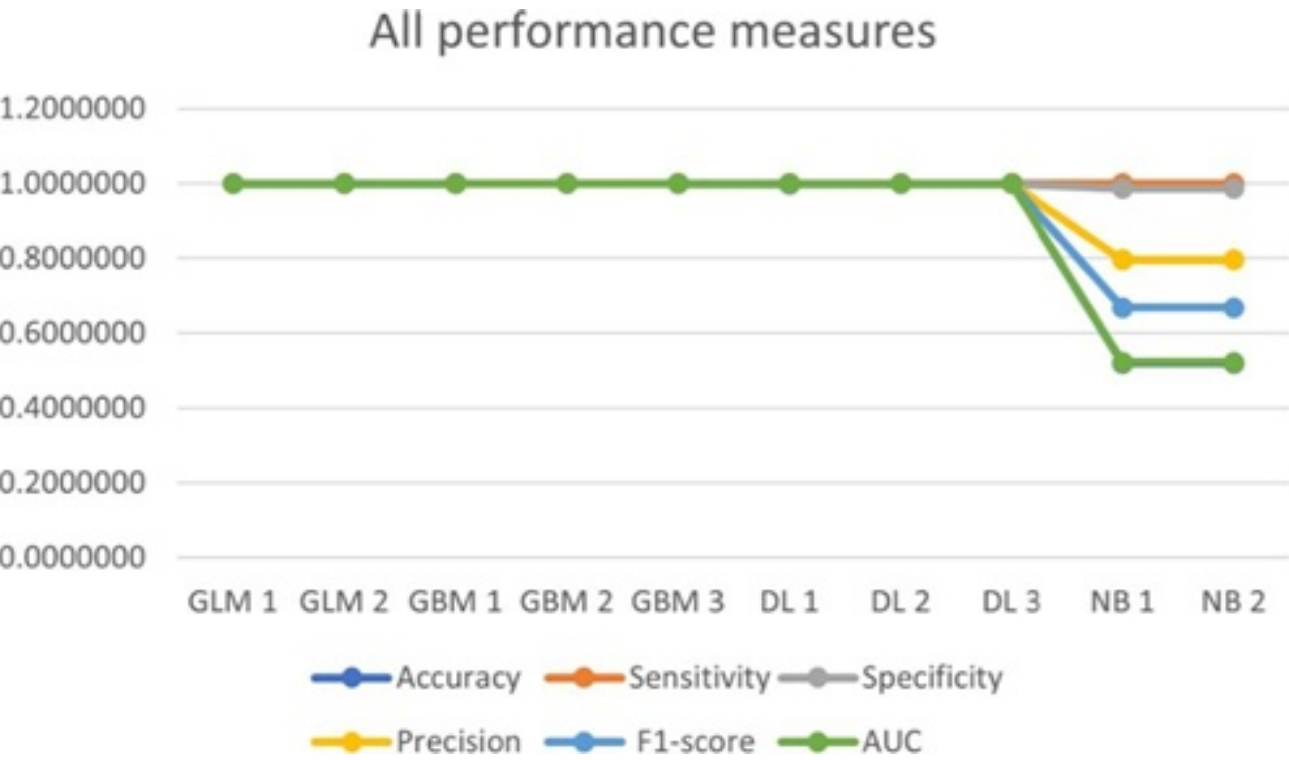
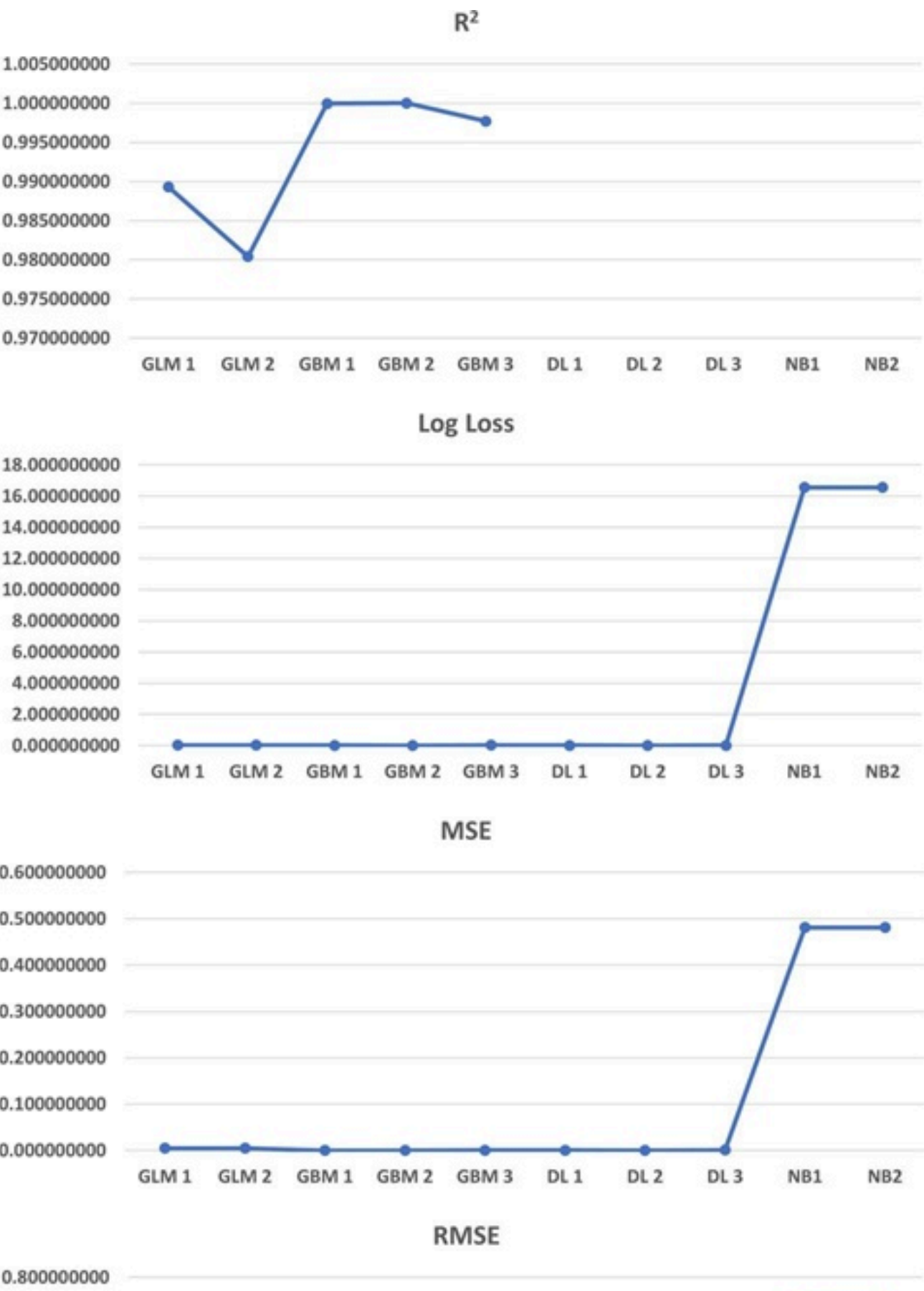
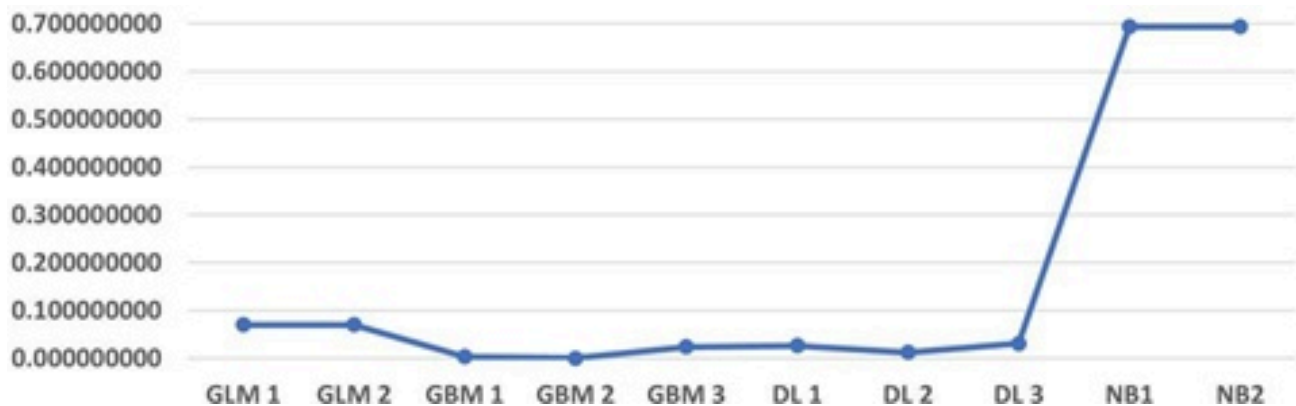


Figure 8. Comparative overview of performance metrics across models: Accuracy (blue), Sensitivity (orange), Specificity (grey), Precision (amber), F1-score (light blue), and Area Under Curve (AUC) (green).





Additionally, NB exhibited markedly higher loss values compared to the remaining models, as illustrated in [Figure 9](#). Overall, these findings indicate that the Naive Bayes classifier is not well-suited to the SGCC dataset used in this study. The following discussion examines the performance of the proposed machine learning approach against the defined evaluation criteria (performance metrics and model losses). It also includes an analysis of training versus validation behavior to evaluate how effectively the selected model fits the SGCC dataset.

Gradient boosting machine algorithm advantages

A key strength of the proposed approach lies in its effective use of finite differences as engineered features within the transformed dataset, combined with the Gradient Boosting Machine's demonstrated ability to detect and leverage these features during preprocessing. The superior results obtained in the GBM experiments align well with the algorithm's established strengths. GBM can be viewed as an advanced form of gradient descent optimization applied to an ensemble of weak learners. Like many powerful algorithms, it offers a range of benefits alongside certain limitations. Among its principal advantages are: (1) the relative ease of achieving very high predictive accuracy; (2) considerable flexibility in adapting to different loss functions through hyperparameter optimization; (3) strong performance on both numerical and categorical features with minimal need for preprocessing; and (4) the ability to natively handle missing values without requiring imputation. On the downside, GBM's iterative error-minimization process can lead to overfitting, particularly when outliers are present, necessitating techniques such as cross-validation to counteract this tendency.

Furthermore, building effective GBM models often requires a large number of trees, which increases both memory consumption and computational time. Finally, the algorithm's high degree of configurability—governed by numerous parameters including the number of iterations, regularization terms, tree depth, and others—demands thorough hyperparameter tuning, typically via extensive grid search. Although interpretability is reduced compared to simpler models, tools such as LIME, partial dependence plots, and feature importance rankings can help provide insights into model behavior.

One of the most valuable aspects of GBM is the significant performance gains achievable through careful hyperparameter tuning. The primary trade-off is the considerable computational effort required for this tuning process. The hyperparameters most frequently adjusted include the number of trees, maximum tree depth, learning rate, and subsampling ratio. In the current study, tuning focused specifically on varying the number of trees, while the remaining hyperparameters were kept at their default values.

Best number of trees

It should be highlighted that the number of decision trees plays a significant role in determining the effectiveness of the Gradient Boosting Machine (GBM) algorithm. In the initial GBM experiment, default settings were applied, resulting in the model being trained with 50 trees—the standard default value. For the second experiment, this number was raised to 500 trees. Notably, early stopping mechanisms were not activated in either of these first two experiments. To prevent potential overfitting caused by the large number of trees, early stopping was enabled in a third experiment, while retaining the

same 500-tree configuration as in the previous run. It is also important to mention that early stopping is not enabled by default for GBM. Among the four machine learning algorithms tested, only Deep Learning automatically uses early stopping.

Given that the GBM models achieved perfect scores across all performance metrics in each of the three experiments, a comparison of model losses becomes essential to assess their relative quality. Although the loss values are extremely small in all cases, clear differences emerge among the three configurations due to variations in key parameters—specifically the number of trees and the inclusion of early stopping. Based on the data presented in **Table 8** and **Figure 5**, it is straightforward to conclude that the parameter settings used in the second experiment deliver the most favorable overall outcome: zero model losses and an R^2 value of exactly 1.0. The next best performance belongs to the first experiment, with the third experiment ranking lowest among the three.

The higher model losses observed in the third experiment—despite matching the 500-tree count of the second—are directly attributable to early stopping during training. Early stopping is governed by three main parameters: the number of stopping rounds, the stopping metric, and the stopping tolerance. An additional parameter, score tree interval, influences evaluation frequency but does not directly control stopping. The stopping rounds parameter specifies the number of consecutive evaluation intervals that must show no improvement (beyond the stopping tolerance) in the chosen stopping metric before training halts. The score tree interval determines how frequently the model is evaluated (every N trees).

In this study, the stopping metric was set to AUC, the stopping rounds to 3, the stopping tolerance to 0.0005, and the score tree interval to 5. These settings are detailed in **Table 2**. As a consequence of these early stopping rules, training terminated after only 30 trees, because the AUC had already reached the perfect value of 1.0 and remained stable across three consecutive evaluation rounds—satisfying the defined conditions. Ultimately, the effective number of trees in the third experiment fell below the default 50 used in the first experiment, which explains the observed pattern in model losses across the trials. Although the differences in loss values are minor, and the losses in the third experiment are only slightly higher than those in the first, the third configuration completed training considerably faster than the other two. This represents a clear benefit, as the 500-tree model in the second experiment incurs a substantially greater computational cost and longer training duration. A trade-off therefore exists between minimizing model losses and reducing training time. Accordingly, the appropriate conclusion is that increasing the number of decision trees reduces model losses (**Figure 10**), but at the cost of greater computational demand and extended training periods. **Figure 11** presents the scoring history from the second GBM experiment, clearly demonstrating a progressive decrease in losses as the number of trees grows, with perfectly overlapping training and validation curves indicating ideal model fit and no signs of overfitting. For comparison, **Figure 12** shows the equivalent scoring trends from the Deep Learning experiments as a function of epoch.

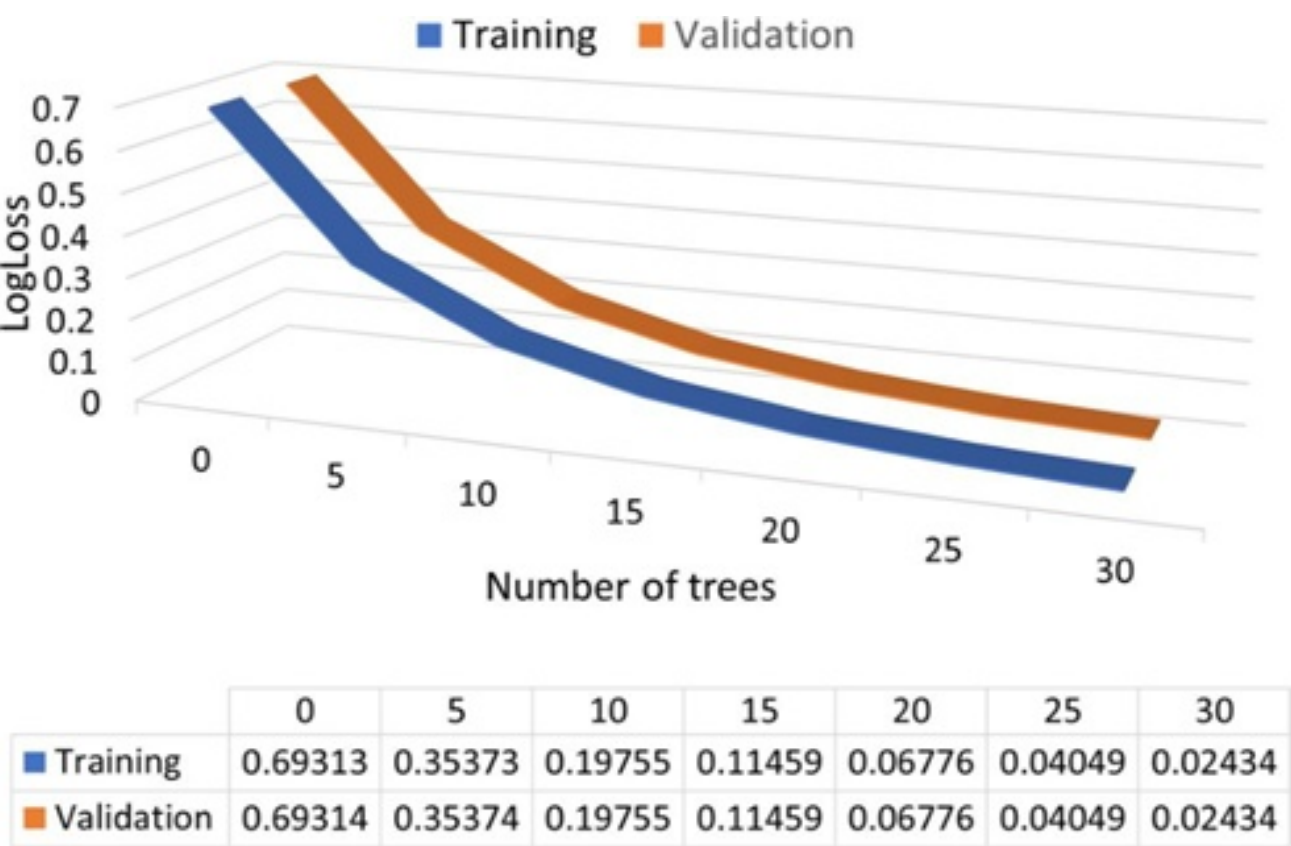
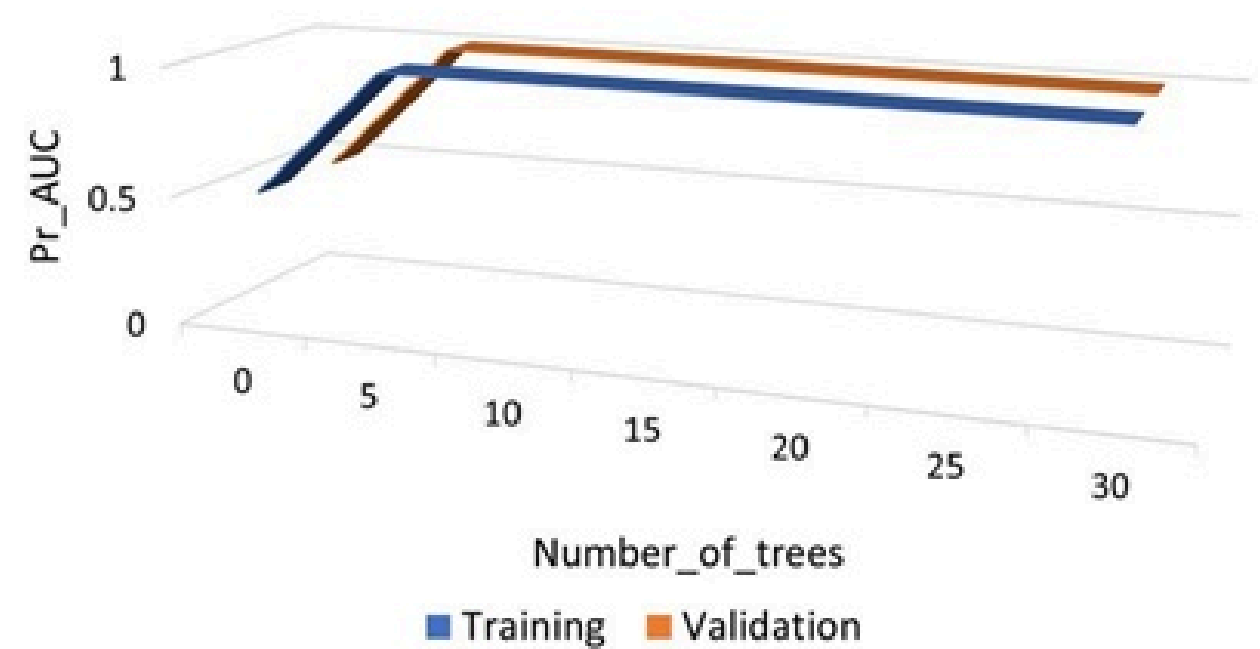
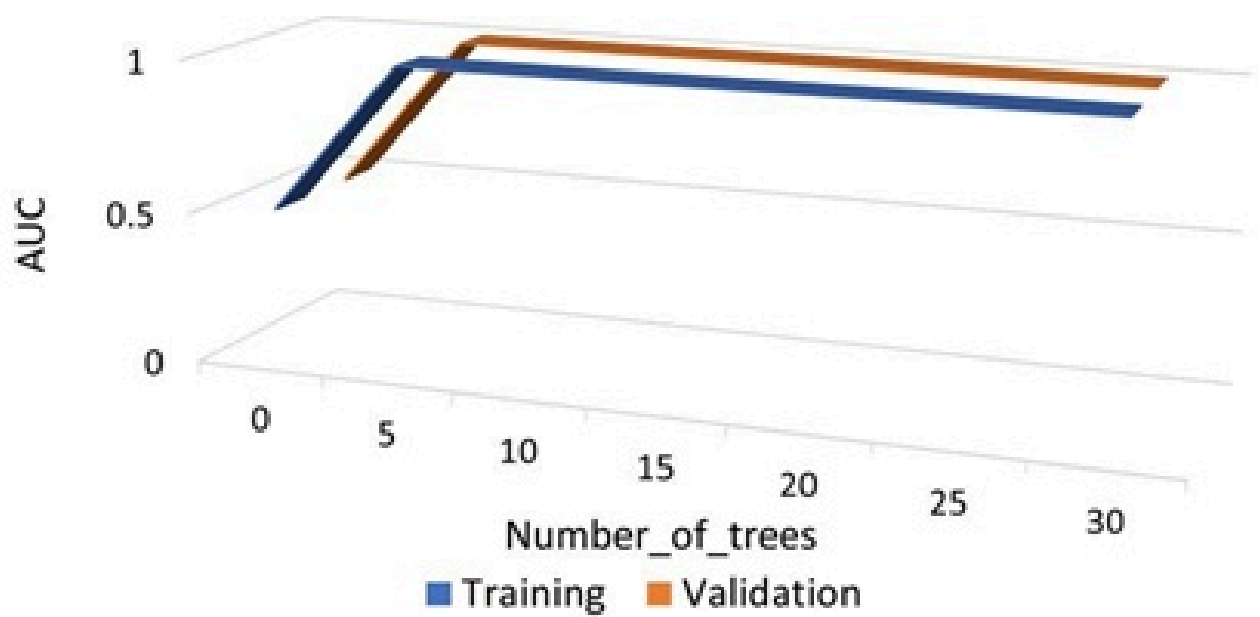
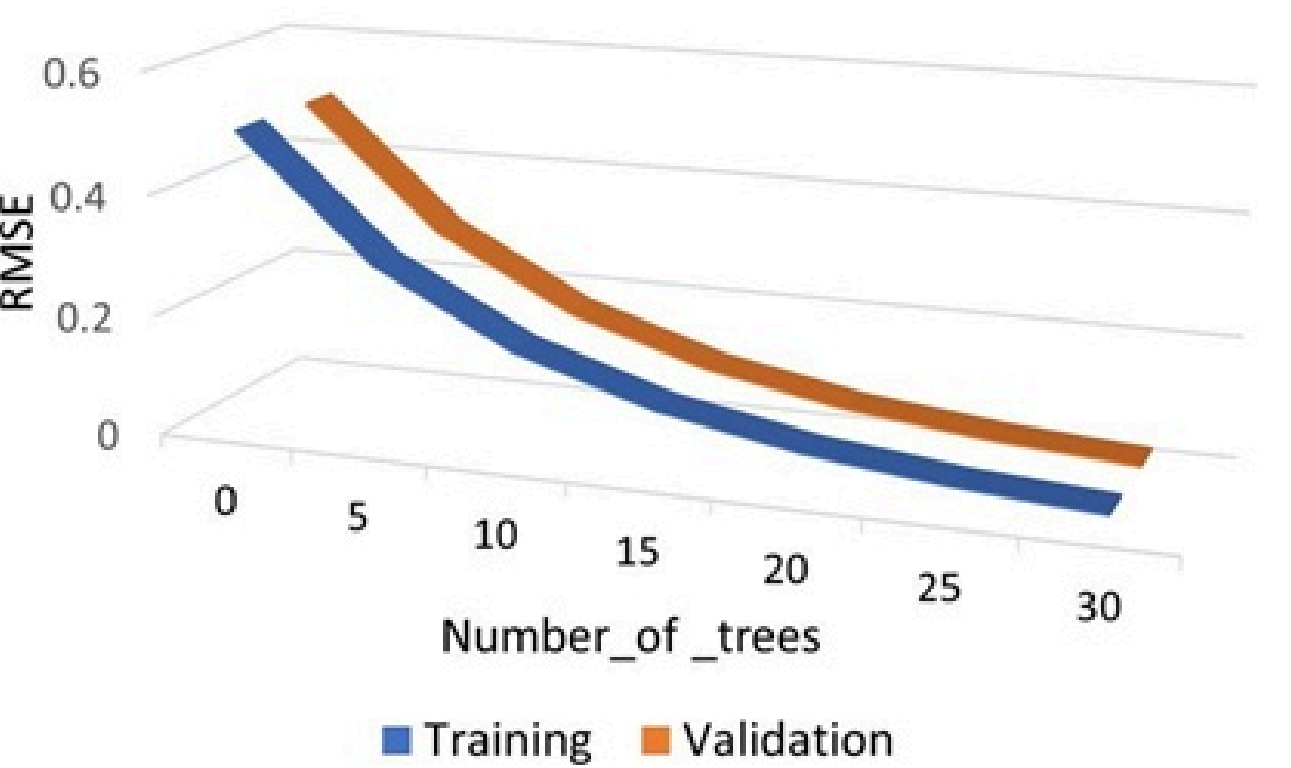
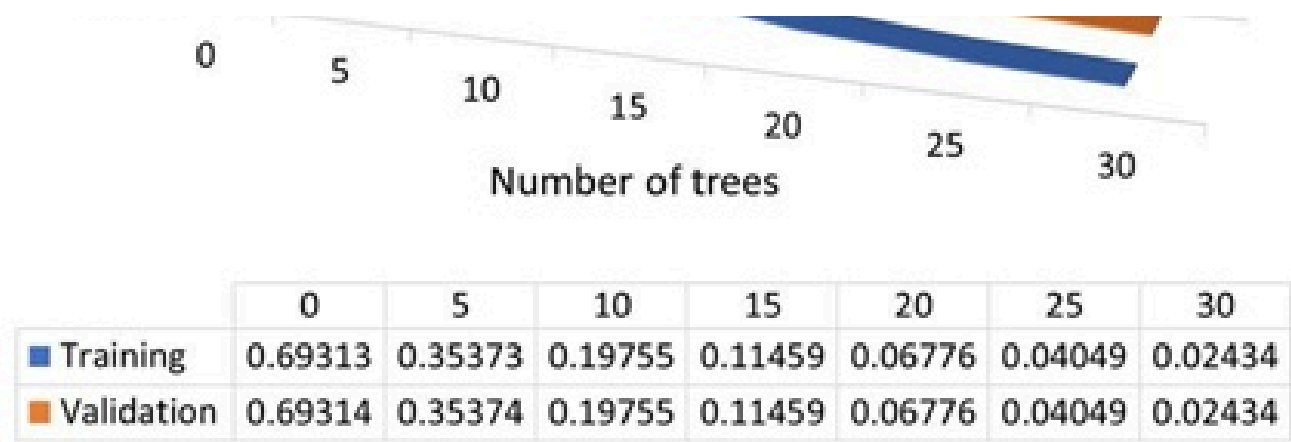
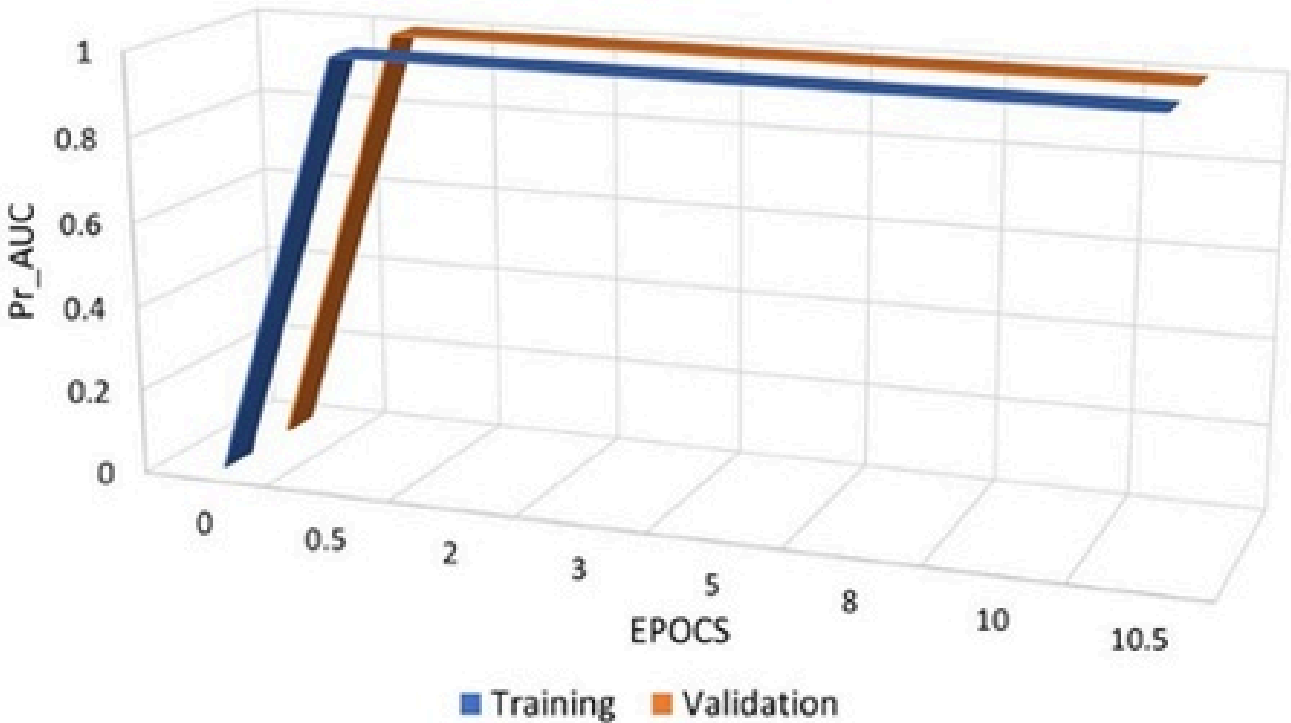
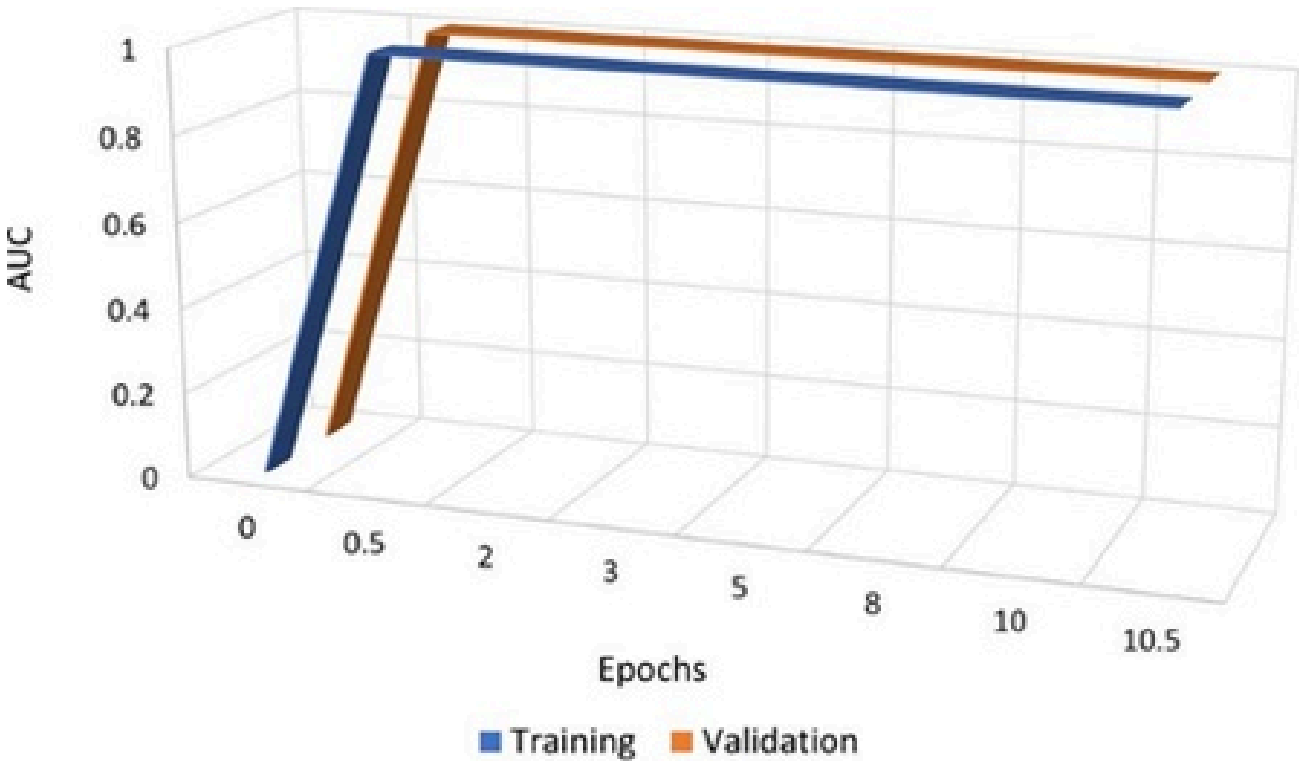


Figure 10. Increasing the number of trees leads to reduced training and evaluation model losses.







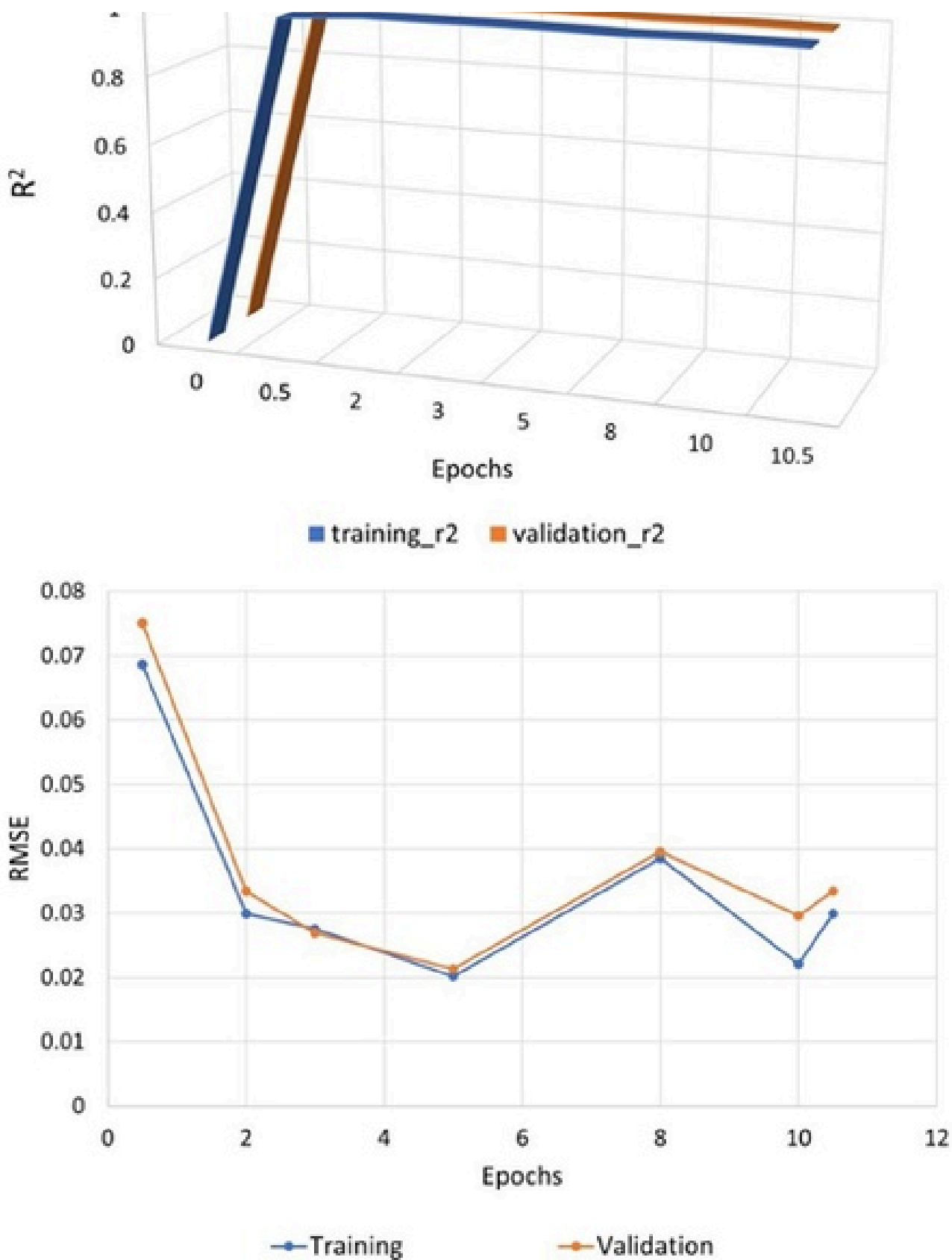


Table 16. Naive Bayes (NB) 2 confusion matrix (vertical: actual; across: predicted).

	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

Ensemble learning

Gradient Boosting Machines (GBMs) are a type of ensemble learning method. The present study highlights the exceptional effectiveness of the GBM model across six key performance metrics, as well as the robustness and flexibility of the proposed NTL fraud detection framework. Given that ensemble techniques were used, three additional experiments were conducted using Distributed Random Forest (DRF) on the transformed SGCC dataset, each with distinct parameter configurations. The first experiment used default settings, in which the algorithm employs 50 trees. In the second experiment, the number of trees was raised to 100. It is generally expected that increasing the number of trees enhances performance in such models. Random Forest algorithms are also widely regarded as relatively resistant to overfitting. A side-by-side comparison of model losses between the GBM and DRF experiments is presented in [Table 17](#).

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

When examining the values in [Table 17](#), the R² scores obtained from the GBM experiments consistently exceed those from the DRF experiments. This difference suggests that the GBM model possesses greater capacity to capture and represent patterns associated with fraudulent behavior. A higher R² value reflects a stronger ability of the model to account for the variability observed in the phenomenon under study.

Comparison with other methods

A wide range of techniques and machine learning algorithms for detecting electricity fraud and non-technical losses (NTLs) have been documented in the existing literature. In direct comparison with these state-of-the-art approaches, the proposed method demonstrates superior results across all six performance indicators. These six metrics provide a comprehensive basis for determining which algorithm performs most effectively on a given dataset, particularly in the domain of NTL detection.

Accuracy reflects the proportion of all instances correctly classified. Recall (also known as sensitivity) quantifies the proportion of actual positive cases (fraud) correctly identified as true positives (TP). Specificity measures the proportion of actual negative cases (non-fraud) correctly identified as true negatives (TN). Precision is the ratio of correctly identified

true positives to the total number of predicted positives. The F1-score serves as a balanced measure that combines recall and precision. Finally, the Area Under the Curve (AUC) yields high values when both recall and specificity are strong.

Table 18 provides a quantitative comparison of the proposed GBM model with leading methods reported in prior studies. The results clearly show that the suggested approach surpasses the benchmarked models across all performance metrics, thereby confirming its superior capability to detect electricity fraud and NTLs. **Figure 13** visually compares the performance of the proposed method (highlighted in blue) with other published approaches, using AUC as the primary metric. The proposed GBM model achieved the highest values for both the Receiver Operating Characteristic AUC (which balances sensitivity and specificity) and the Precision-Recall AUC (PR-AUC).

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

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	

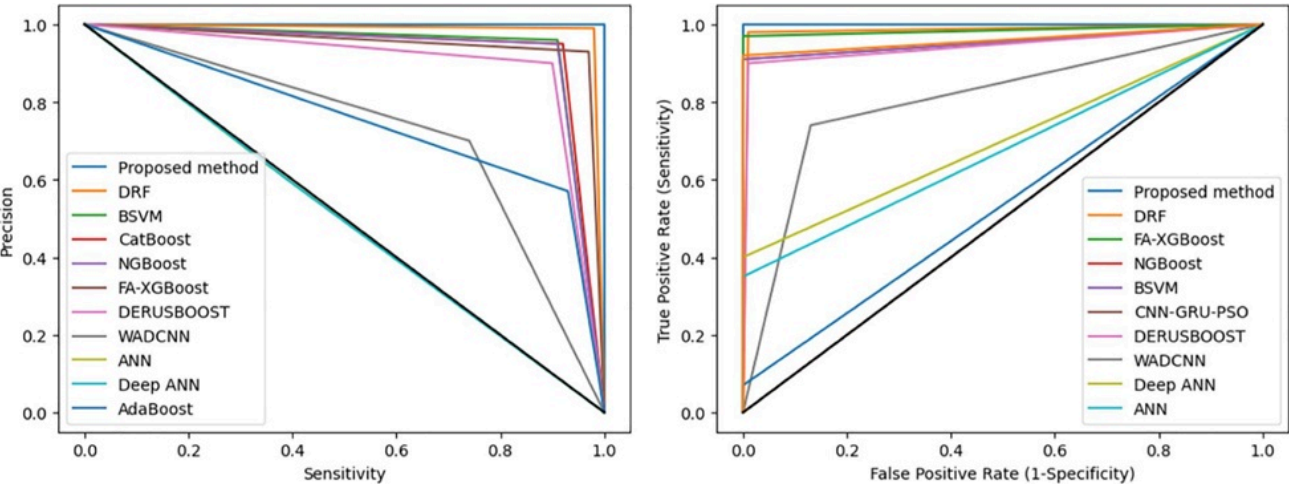


Figure 13. Performance comparison of the proposed method against state-of-the-art approaches in terms of PR_AUC (left panel) and ROC-AUC (right panel).

Future work

The current research evaluated four machine learning models across 10 experiments (ranging from 2 to 3 per model) and used the Boruta algorithm to select features from 72 columns. The objective was to retain the most informative features while eliminating irrelevant or constant ones, ultimately yielding a promising two-stage framework for detecting non-technical losses in smart meters arising from electricity theft and fraud.

Future investigations are planned with several directions in mind. First, the model will be tested on a broader range of smart grid datasets to verify its generalizability across diverse consumption patterns and reduce dependence on any single dataset. Second, ablation studies will be conducted to quantify the individual contributions of the feature engineering stage and the choice of machine learning algorithms. In summary, this work proposes a practical, highly accurate two-step methodology that integrates sophisticated feature engineering with recent advances in machine learning to effectively address this critical challenge.

Conclusion

Non-technical losses, frequently caused by electricity theft and fraudulent activities, impose substantial financial burdens on utility providers, often totaling millions of dollars annually. The present study proposes an innovative and practical framework for identifying NTLs and electricity fraud within smart grid environments. A standout feature of the recommended GBM model is its ability to consistently achieve performance scores exceeding 0.99 across multiple metrics, underscoring its high reliability and stability. As an ensemble method, it constructs multiple decision trees—each trained differently—and aggregates their predictions to produce highly accurate classifications.

The proposed methodology follows a two-stage process. In the first stage, abrupt changes or sudden deviations in consumption are detected by computing the sum of finite differences around each point, with further refinement achieved through the application of ARIMA and Holt-Winters models. Subsequently, relevant features are extracted from the smart meter time-series data in the SGCC dataset. A machine learning classifier then processes the transformed dataset to label each meter as either fraudulent or legitimate. The incorporation of finite differences enables the detection of sharp discontinuities in usage patterns, while time-series feature extraction enhances the dataset's suitability for machine learning analysis. The experimental results clearly demonstrate the effectiveness of this approach. Compared to existing techniques, the proposed method delivers improved performance in precision, accuracy, AUC, specificity, recall, and F1-score, confirming its value as an advanced solution for non-technical loss detection in modern power systems.

Acknowledgements

None

Conflict of interest

None

Financial support

None

Ethics statement

None

Received: 05 Apr 2024

Revised: 17 May 2024

Accepted: 29 Jun 2024

Published online: 20 September 2024

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

- Haq E, Khan I, Khalid S, Ahmed S, Hussain T, Ali A. Electricity-theft detection for smart grid security using smart meter data: a deep-CNN based approach. *Energy Rep.* 2023;9:634–43.
<https://doi.org/10.1016/j.egy.2022.11.072>.
- Zheng Z, Yang Y, Niu X, Dai HN, Zhou Y. Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids. *IEEE Trans Ind Inf.* 2018;14(4):1606–15.
<https://doi.org/10.1109/tii.2017.2785963>.
- Hasan M, Toma RN, Nahid AA, Islam MM, Kim JM. Electricity theft detection in smart grid systems: a CNN-LSTM based approach. *Energies.* 2019;12(17):3310.
<https://doi.org/10.3390/en12173310>.
- Nagi J, Yap KS, Tiong SK, Ahmed SK, Mohamad M. Nontechnical loss detection for metered customers in power utility using support vector machines. *IEEE Trans Power Deliv.* 2010;25(2):1162–71.
<https://doi.org/10.1109/tpwrd.2009.2030890>.
- Glauner P, Boechat A, Dolberg L, State R, Bettinger F, Rangoni Y. The challenge of non-technical loss detection using artificial intelligence: a survey. *arXiv:1606.00626*. 2016.
- Saeed MS, Mustafa MW, Sheikh UU, Jumani TA, Mirjat NH. Detection of non-technical losses in power utilities—a comprehensive systematic review. *Energies.* 2020;13(18):4727.
<https://doi.org/10.3390/en13184727>.
- Badrinath Krishna V, Iyer RK, Sanders WH. ARIMA-based modeling and validation of consumption readings in power grids. In: *Proceedings of the International Conference on Critical Information Infrastructures Security*. Berlin: Springer; 2015. p. 199–210.
- Xia X, Xiao Y, Liang W. A suspicion assessment-based inspection algorithm to detect malicious users in smart grid. *IEEE Trans Inf Forensics Secur.* 2020;15:361–74.
<https://doi.org/10.1109/tifs.2019.2921232>.
- Viegas JL, Vieira SM, Sousa JMC, Melo P, Ferreira J. Solutions for detection of non-technical losses in the electricity grid: a review. *Renew Sustain Energy Rev.* 2017;80:1256–68.
<https://doi.org/10.1016/j.rser.2017.05.193>.

Jokar P, Arianpoo N, Leung VC. Electricity theft detection in AMI using customers' consumption patterns. *IEEE Trans Smart Grid*. 2016;7(1):216–26.

<https://doi.org/10.1109/tsg.2015.2425222>.

Guo Y, Ten CW, Jirutitijaroen P. Online data validation for distribution operations against cyber tampering. *IEEE Trans Power Syst*. 2014;29(2):550–60.

<https://doi.org/10.1109/tpwrs.2013.2282931>.

Messinis GM, Hatziaargyriou ND. Review of non-technical loss detection methods. *Electr Power Syst Res*. 2018;158:250–66.

<https://doi.org/10.1016/j.epsr.2018.01.005>.

Ferreira TSD, Trindade FC, Vieira JC. Load flow-based method for nontechnical electrical loss detection and location in distribution systems using smart meters. *IEEE Trans Power Syst*. 2020;35(5):3671–81.

<https://doi.org/10.1109/tpwrs.2020.2981826>.

Tariq M, Poor HV. Electricity theft detection and localization in grid-tied microgrids. *IEEE Trans Smart Grid*. 2018;9(3):1920–9.

<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; 2011. Vol. 2, p. 413–6.

McLaughlin S, Podkuiko D, Mirosavljevic S, Sezer S. A multi-sensor energy theft detection framework for advanced metering infrastructures. *IEEE J Sel Areas Commun*. 2013;31(7):1319–30.

<https://doi.org/10.1109/jsac.2013.130714>.

Xiao Z, Xiao Y, Du DHC. Exploring malicious meter inspection in neighborhood area smart grids. *IEEE Trans Smart Grid*. 2013;4(1):214–26.

<https://doi.org/10.1109/tsg.2012.2229397>.

Angelos EWS, Saavedra OR, Cortés OAC, Souza AN. Detection and identification of abnormalities in customer consumptions in power distribution systems. *IEEE Trans Power Deliv*. 2011;26(4):2436–42.

<https://doi.org/10.1109/tpwrd.2011.2161621>.

Zheng K, Chen Q, Wang Y, Kang C, Xia Q. A novel combined data-driven approach for electricity theft detection. *IEEE Trans Ind Inf*. 2019;15(3):1809–19.

<https://doi.org/10.1109/tii.2018.2873814>.

Ramos CCO, Souza AN, Chaves GF, Nascimento VH, Vitorino IM. Identification and feature selection of nontechnical losses for industrial consumers using the software WEKA. In: *Proceedings of the 2012 10th IEEE/IAS International Conference on Industry Applications*. Fortaleza, Brazil; 2012. p. 1–6.

Khan ZA, Adil M, Javaid N, Saqib MN, Shafiq M, Ali A. Electricity theft detection using supervised learning techniques on smart meter data. *Sustainability*. 2020;12(19):8023.

<https://doi.org/10.3390/su12198023>.

Depuru SSSR, 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; 2011. p. 1–8.

Nagi J, Yap KS, Tiong SK, Ahmed SK, Mohamad M. 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; 2008. p. 907–12.

Nizar A, Dong Z, Wang Y. Power utility nontechnical loss analysis with extreme learning machine method. *IEEE Trans Power Syst*. 2008;23(3):946–55.

<https://doi.org/10.1109/tpwrs.2008.926431>.

Costa BC, Alberto BL, Porto JM, Ferreira VH. Fraud detection in electric power distribution networks using an ann-based knowledge-discovery process. *Int J Artif Intell Appl*. 2013;4(6):17–23.

<https://doi.org/10.5121/ijaia.2013.4602>.

Hussain S, Mustafa MW, Jumani TA, Baloch SK, Memon ZA, Shah SF. A novel feature engineered-CatBoost-based supervised machine learning framework for electricity theft detection. *Energy Rep*. 2021;7:4425–36.

<https://doi.org/10.1016/j.egyr.2021.07.008>.

Hussain S, Mustafa MW, Jumani TA, Baloch SK, Memon ZA, Shah SF. A novel feature-engineered-NGBoost machine-learning framework for fraud detection in electric power consumption data. *Sensors*. 2021;21(24):8423.

<https://doi.org/10.3390/s21248423>.

Khan IU, Javaid N, Ahmad A, Khan ZA, Alrajeh N, Ali A. Big data analytics for electricity theft detection in smart grids. In: *Proceedings of the 2021 IEEE Madrid PowerTech*. Madrid, Spain; 2021. p. 1–6.

Bohani FA, Khan SA, Khan I, Habib M, Ahmad J. Comprehensive analysis of supervised learning techniques for electricity theft detection. *J Electr Comput Eng*. 2021;2021:9136206.

Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*. 2014.

H2O.ai. Generalized linear model (GLM) [Internet]. H2O.ai Documentation; 2024 [cited 2023 Jul 25]. Available from:

<https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/glm.html>

H2O.ai. Gradient boosting machine (GBM) [Internet]. H2O.ai Documentation; 2024 May 31 [cited 2023 Jul 25]. Available from:

<https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/gbm.html>

Click C, Malohlava M, Candel A, Roark H, Parmar V. Gradient boosting machine with H2O. H2O.ai; 2017.

Alarfaj FK, Malik I, Khan HU, Almusallam N, Ramzan M, Ahmed A. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access*. 2022;10:39700–15.

<https://doi.org/10.1109/access.2022.3166891>.

Yao D, Wen M, Liang X, Fu Z, Zhang K, Yang B. Energy theft detection with energy privacy preservation in the smart grid. *IEEE Internet Things J*. 2019;6(5):7659–69.

<https://doi.org/10.1109/jiot.2019.2903312>.

Muniz C, De Souza AN, Da Silva IN, Lacerda W. 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; 2009. p. 2176–82.

Depuru SSSR, 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; 2012. p. 1–6.

Aiello S, Kraljevic T, Maj P, Nocera F, Pappalardo L. Machine Learning with R and H2O. H2O.ai booklet; 2016.

Badawi SA, Alghamdi N, Alhassan AM, Alqahtani A, Al-Naffakh T. A novel time-series transformation and machine-learning-based method for NTL fraud detection in utility companies. *Mathematics*. 2022;10(11):1878.

<https://doi.org/10.3390/math10111878>.

Kulkarni Y, Kulkarni A, Nimbalkar P, Patil S. EnsembleNTLDetect: an intelligent framework for electricity theft detection in smart grid. In: *2021 International Conference on Data Mining Workshops (ICDMW)*. IEEE; 2021. p. 527–36.

Mujeeb S, Javaid N, Ahmed A, Khan ZA, Ali A, Iqbal M. An efficient electricity theft detection scheme with additive communication layer. In: Proceedings of the ICC 2020—2020 IEEE International Conference on Communications. Dublin, Ireland; 2020. p. 1–6.

Ullah A, Javaid N, Ahmed A, Khan ZA, Ali A, Iqbal M. 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; 2020. p. 1598–602.