# Data-Driven Insights: Boosting Algorithms to Uncover Electricity Theft Patterns in AMI

Inam Ullah Khan[1], Arshid Ali[2], C. James Taylor[3], and Xiandong Ma[3]
[1]*Lyle School of Engineering, Southern Methodist University, Dallas, USA*
[2]*Department of Electrical Engineering and Computer Science, South Dakota State University, USA*
[3]*School of Engineering, Lancaster University, Bailrigg, Lancaster LA1 4YW, UK*

***Abstract*—This study introduces a sophisticated supervised machine learning method for electric theft detection utilizing a customized Histogram Gradient Boosting (HGB) algorithm. Comprehensive preprocessing, including imputation, normalization, outlier management, and resampling, ensures the time-series data is accurately prepared for analysis. The SMOTE-ENN algorithm corrects class imbalances, preparing the data for the feature optimization stage, in which key features are selected and extracted. The HGB algorithm, enhanced through Bayesian optimization, is central to the training process, resulting in a model that precisely classifies electricity consumption patterns as genuine or fraudulent. The robustness of the model is evaluated against other recognized boosting methods, such as Adaptive Boosting (ADB), Gradient Boosting Decision Tree (GBDT), and LightGBM, alongside various ensemble and traditional machine learning models. Utilizing key performance metrics like accuracy, F1 score, and AUC for validation, the proposed model yields very promising results, with 93% accuracy, 95% F1 score, and 98% AUC, outperforming the comparison group under similar dataset and hyperparameter conditions. This underscores the model's potential as a highly accurate tool for combating electricity theft within an advanced metering infrastructure (AMI).**

***Index Terms*—Electricity Theft Detection, Class Balancing, Feature Engineering, Boosting Algorithms, Advanced Metering Infrastructure, Smart Grid.**

## I. INTRODUCTION

### A. *Background*

As we move towards a future in which cutting-edge technology makes cities more connected and effective, the idea of smart grids becomes increasingly important [1]. These grids, enhanced with digital technology, are set to change the way energy is used, managed, and distributed in tomorrow's urban landscapes. Nevertheless, the futuristic concept of interconnected energy systems faces various obstacles, one of which is the issue of technical and non-technical losses (NTL) [2]. Technical losses, inherent to every electrical system, result from the dissipation of energy during the process of transmitting and distributing it. The aforementioned losses, including transformer losses, corona discharge, and resistance in wires, can be reduced by the use of technological advancements and improvements in the system [3]. NTL, which can result from different irregularities such as energy theft, inaccuracies in meter parameterization, installation problems, or defective meters, provide a significant burden for power companies [4]. These losses not only lead to substantial decreases in revenue but also introduce uncertainty into the functioning of the power system by concealing true consumption patterns [5]. The mitigation of NTL is important to energy providers, as these losses constitute a significant proportion (40-60%) of the overall power losses [6].

Energy theft, central to NTL, poses a severe challenge to the smart grid vision and the protection of the overall energy infrastructure [7]. It has major financial consequences, with estimated revenue losses of $96 billion annually, undermining the economic and operational stability of global energy infrastructures. In the US alone, energy theft-driven NTL costs amount to $9 billion. The issue is not isolated; Canada reports losses of $100 million, the UK $234 million, and the State Grid Corporation of China (SGCC) close to $1.8 billion [8]. Illegitimate acts exceed mere financial implications. They impact on energy management protocols, threaten equipment integrity, and interrupt seamless power flow. When energy is stolen, grids experience unexpected loads, posing challenges in energy distribution and increasing risks of outages.

The advent of advanced metering infrastructure (AMI) has facilitated the development of novel approaches to detect NTL. This task was previously difficult due to the coarse granularity of the data [6]. Utilities now have access to frequent and precise measurements of energy consumption due to the widespread deployment of smart meters. This granular data provides greater insight into consumers' behaviour, improving the ability to identify irregularities [9]. The sheer scale of data generation, 22 GB daily from approximately 2.2 million smart meter users, demands robust solutions for storage and new analytics. Although this wealth of data is crucial for improving energy theft detection, customer service, and operational efficiency, the path forward is laden with obstacles, including ensuring data integrity, maintaining privacy, and scaling systems to meet demand [10].

In addition to NTL, other fraudulent techniques pose significant challenges to smart grid and smart metering applications [11]. These techniques include meter tampering, where individuals physically interfere with meters to alter readings, either by slowing down the meter or stopping it altogether. Another common method is bypassing meters, where illegal connections are made to the power supply ahead of the meter, allowing consumption of electricity without recording usage. Furthermore, false meter readings can be transmitted by manipulating the data from the meter to report lower consumption. These fraudulent activities undermine the integrity and

reliability of smart grids, again leading to substantial financial losses and operational disruptions.

AMIs raises critical privacy and security issues that need to be addressed to protect consumer information [12]. Smart meters collect detailed data on energy usage, which can inadvertently reveal personal information, such as living habits and occupancy patterns. For instance, detailed consumption data can indicate when residents are home, their daily routines, and appliance usage patterns, which can be misused if not adequately protected. Additionally, energy usage patterns can reveal when a home is occupied or vacant, posing security risks if accessed by unauthorized parties. Ensuring the integrity and confidentiality of data is required to maintain consumer trust.

AMIs have significant implications for consumer households, offering both benefits and challenges. Smart meters enable consumers to monitor their energy usage in real-time, facilitating better energy management and cost savings. They also support demand response programs, where consumers can adjust their usage during peak times in response to price signals, and improve billing accuracy by reducing errors associated with manual readings [13]. However, there are challenges, such as privacy concerns, the cost of implementing and maintaining smart meters, and potential technical issues related to installation and operation. Addressing these challenges through robust security measures, consumer education, and support from utilities is essential for realizing the full benefits of smart metering technology.

### *B. **Related Works***

NTL detection approaches diverge into theoretical, hardware-based, and non-hardware-based methodologies [14]–[16]. Theoretical methods correlate socio-economic and demographic data to inform policy-making for NTL mitigation [17]. Hardware-based strategies employ physical devices like sensors and detection equipment to monitor electrical parameters, triggering alerts upon tampering attempts. However, these incur high costs for installation and upkeep, limiting their utility for many power companies [18]. Non-hardware-based methods, which avoid these costs, bifurcate into game-theoretic and data-driven models [19], [20]. Game-theoretic approaches frame detection as a strategic interaction between utility providers and thieves, but complexity in defining the roles and strategies of involved parties hampers practical application [21]. Data-driven methods, on the other hand, leverage consumer usage data and are subdivided into unsupervised learning [22], which clusters consumer profiles based on usage patterns, and supervised learning [3], which relies on labeled data to train algorithms to distinguish between legitimate and fraudulent usage. The present article focuses on supervised learning, which faces challenges such as managing missing data, addressing class imbalance, feature selection, classifier optimization, and model interpretation [23], [24].

Smart meter datasets are frequently plagued by inconsistencies, with null values being a common occurrence. These issues often arise from a range of factors, including equipment malfunctions, errors in data estimation, ad hoc repairs, and storage anomalies. Such data irregularities present substantial challenges for machine learning classifiers tasked with identifying consumption patterns. A critical review of 34 papers on theft detection using supervised machine learning, as noted in [25], reveals that a mere 50% have tackled the issue of missing data, a nontrivial concern that can significantly skew analytical outcomes. To address this, the literature suggests a variety of data imputation methods. Among these, the most prevalent practices remain the deletion of missing entries or the substitution of null values with the mean of adjacent data points, as documented in [26], [27]. While these methods are straightforward and widely used, they are not without drawbacks. Deletion can lead to a considerable loss of valuable information, potentially biasing the results.

Class imbalance in labeled datasets significantly impedes machine and deep learning models, often introducing a bias towards the majority class and consequently neglecting crucial minority classes, such as actual theft cases. Rectifying this imbalance is imperative to maintain the efficacy and fairness of data-driven models, especially in detecting NTL. Inam et al. [28] and Gunturi and Sarkar [29] have employed the Synthetic Minority Over-sampling Technique (SMOTE) to enhance minority class representation with a degree of success. However, SMOTE's tendency to randomly oversample can lead to model overfitting and reduced generalizability. Conversely, Buzau et al. [30] explored under-sampling, removing samples from the majority class to achieve balance. While straightforward, such techniques risk the loss of significant information, potentially compromising the model's accuracy.

Feature engineering is pivotal in supervised NTL detection, yet existing literature often presents a piecemeal approach to feature selection and extraction procedures. Studies like those of Simona *et al.* [31] survey feature engineering methods extensively but fall short of integrating selection with extraction, potentially limiting model performance. Razavi *et al.* [24] explore Genetic Programming for feature construction, but scalability remains a concern. The FRESH algorithm, employed by Saddam et al. [32], effectively identifies relevant features but may increase model complexity. Similarly, Darshana *et al.* [33] use a gradient boosting-based Weighted Feature Importance (WFI) model for feature elimination, risking the oversight of predictive feature interactions. Shoaib *et al.* [34] introduce innovative feature engineering methods, yet they often require extensive computational resources, raising concerns about scalability and efficiency. Pamir *et al.* [35] apply autoencoders for feature extraction from historical data, trading off interpretability for sophistication. These studies, while forward-moving, accentuate the necessity for a unified approach that balances computational efficiency with the precision and interpretability of models.

Algorithm selection plays a crucial role in determining the effectiveness of classification outcomes in NTL. In [36], the authors apply metaheuristic approaches, specifically artificial bee colony and genetic algorithms, paired with denoising au-

toencoders, to enhance feature selection, with Support Vector Machine (SVM) as the classifier. While this model achieved an AUC of 90% when applied to an SGCC electricity consumption dataset, it incurred high computational costs, particularly due to the extensive tuning required for SVM hyperparameters, such as the cost penalty ($C$), loss function parameter ($\epsilon$), and kernel parameter ($\gamma$). Another notable approach employed a supervised Categorical Boosting (CatBoost) model with SMOTE-Tomek resampling for NTL detection on the SGCC dataset [32]. This method demonstrated strong performance across multiple metrics (e.g., 93% accuracy, 0.87 MCC) but did not include AUC evaluation, a critical metric for highly imbalanced datasets. Additionally, CatBoost's ordered boosting, though effective in reducing overfitting, added computational demands, resulting in slower training.

Researchers have also investigated other classifiers, including Decision Trees (DT) [37], Random Forest (RF) [38], and Gradient Boosting Decision Trees (GBDT) [39], to achieve optimal classification results. However, conventional data-driven approaches often face significant challenges, such as overfitting, where models perform well on training data but struggle to generalize effectively to unseen data. Furthermore, the high dimensionality of these datasets introduces additional computational complexity, potentially leading to the curse of dimensionality and impairing model performance. The constantly evolving nature of electricity theft tactics further necessitates ongoing updates and refinements in detection algorithms to maintain effectiveness.

### C. *Contributions*

This article concerns a supervised learning scenario for the analysis of time-series electricity consumption data. The proposed approach is applied to the SGCC daily consumption dataset alluded to above. These data are tagged with labels denoting normal or abnormal usage, and are available to researchers following prior work by [19]. The labels are derived from expert analysis and historical instances of confirmed electricity theft, hence serve as a foundation for the training and evaluation of new models and algorithms. Fig. 1 shows the consumption patterns for an illustrative honest and dishonest consumer [19]. Typically, as in Fig. 1, honest consumers exhibit a lower, more stable consumption pattern.

We propose a sequential framework anchored by the Histogram Gradient Boosting (HGB) algorithm. HGB iteratively refines decision trees, with each iteration designed to ameliorate the errors of its forerunner. Its innovative binning strategy divides feature values into discrete bins, which not only aids in accurately capturing the data distributions but also enhances computational speed. Although HGB is a promising approach, the following challenges must be addressed:

- ***Sensitivity to Noisy Data:*** Despite its ability to handle a wide range of data, HGB's architecture is susceptible to data noise. Its iterative refining method, which is intended to gradually enhance the system, might accidentally increase the impact of noise and outliers. As a result, this could lead to inaccurate prediction outcomes. Although error correction procedures are intended to refine predictions, they can occasionally amplify the anomalies inherent in noisy datasets.
- ***Complexity in High-Dimensional Data:*** The binning process of HGB is designed to handle continuous features efficiently. However, the model faces interpretability issues when deals high-dimensional data. With an increase in feature dimensionality, the number of corresponding bins increases. This can veil the relationships between features and their predictive impacts, complicating the model's interpretability for classification problems.
- ***Class Imbalance Challenge:*** The HGB algorithm's learning ability is compromised in situations when training data has a high-class imbalance issue. A disproportionate data class distribution can bias the algorithm towards the majority class (honest) instances, diminishing its efficacy in learning and identifying patterns pertinent to the minority class (theft) instances. The trained models may perform well on the training data but falter on new data, neglecting the nuances of minority classes.

To address these issues, the present work introduces a comprehensive, sequential framework designed to detect electricity theft within large-scale datasets. This framework is based on three components: data preprocessing, feature engineering, and optimized classification, as summarized in Fig. 2. During the initial phase of data preprocessing, we employ a suite of algorithms that preserve the temporal integrity of the data. The feature engineering process is designed to select those features that are relevant to the temporal order of the data points and for dimensionality reduction of the electricity consumption record. The aim is to simplify the dataset by reducing the number of features while still retaining the important time-related information that could affect electricity consumption fluctuations over time. This systematic preparation sets the stage for the application of a boosting classifier.

The main contributions of this paper are summarised below:

1) We present an integrated electricity theft detection framework to make accurate data-driven predictions in smart grids. To our knowledge, this is the first time data preprocessing, feature selection, extraction and classification are all integrated in this manner for the studied problem.
2) The preprocessing stage employs data imputation techniques to rectify missing values, followed by robust scalar and quantile transformer algorithms for dataset standardization. We harness the SelectKBest (SKB) algorithm to identify and select key features with high predictive performance. To contend with the high dimensionality, Principal Component Analysis (PCA) is utilized to maintain data tractability while preserving its intrinsic informational richness. For a balanced representation of both minority and majority instances, we adopt the SMOTE-Edited Nearest Neighbors (SMOTE-ENN). At the core of the proposed methodology is the HGB classifier. HGB performance is further improved with hyperparameter tuning using Bayesian optimisation.

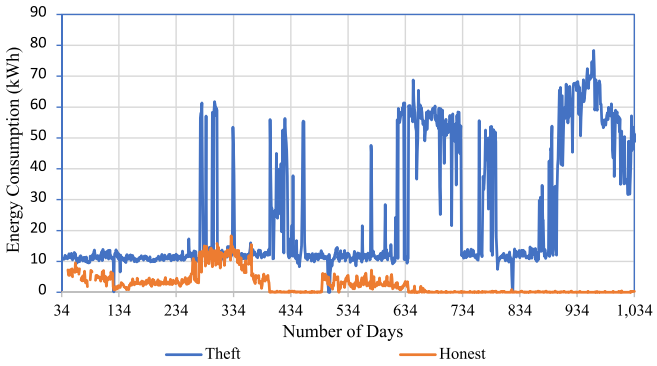


Fig. 1: Daily Electricity Consumption for an illustrative Honest and Theft Consumer (Data source: State Grid Corporation of China (SGCC) electricity consumption records, 2015-2019)

3) The framework's effectiveness is showcased through several experiments with real-world grid data. Numerical results suggest that our proposal has superior performance when compared to other state-of-the-art methods.

The structure of this paper as follows: Section II describes the proposed methodology, beginning with data preprocessing, advancing through feature engineering, and culminating with classification. Section III presents empirical results and the subsequent analysis. Finally, Section IV discusses key findings, wider implications and potential applications.

## II. PROPOSED METHODOLOGY

Algorithm 1 and Fig. 2 illustrates the architecture of our NTL detection framework, iHGB, which is structured into three key phases: initial data pre-processing, detailed feature engineering, and integrated models training and testing, followed by interpretation.

### A. *Data Preprocessing*

This phase is segmented into the following tasks: handling missing data, data distribution transformation for optimum representation, data scaling and data resampling to ensure its congruence with analytical requirements.

*1) **Handling Missing Data**:* Our preprocessing approach addresses the missing data within the original SGCC dataset, where such gaps account for 25% of the information. We implement a straightforward imputer strategy [5]. This method involves the integration of the mean values of existing data, for a given consumer, into the places where 'NaN' (Not a Number) entries occur, thus ensuring data continuity and integrity:

$$\hat{x_i} = \frac{1}{n}\sum_{j=1}^{n} x_j \tag{1}$$

Here, $\hat{x_i}$ symbolizes the inferred value for the absent datum $i$, with $x_j$ denoting the observed value at datum point $j$, and $n$ indicating the aggregate of observed data points.

*2) **Mitigating Outliers**:* Following interpolation, we employ a robust scaler for data normalization. This scaler effectively mitigates distortions caused by outliers by utilizing the interquartile range, thus providing greater resilience against outliers compared to methods based on mean and variance [40]. The transformation of the data value $x_i$ to its scaled counterpart $\hat{x_i}$ is as follows:

$$\hat{x_i} = \frac{x_i - Q_1(x)}{Q_3(x) - Q_1(x)} \tag{2}$$

Within this formulation, $Q_1(x)$ and $Q_3(x)$ stand for the first and third quartiles of attribute $x$, respectively.

*3) **Data Scaling**:* Our preprocessing pipeline includes data scaling using a quantile transformation (QT) [41], which normalizes the distribution of the data to approximate a Gaussian distribution, facilitating compatibility with subsequent classification algorithms. This transformation corrects for skewness and kurtosis, potentially improving classifier accuracy. The transformation for a given data point $x_i$ in feature $x$ is mathematically expressed as:

$$\hat{x_i} = Q\left(\frac{\text{rank}(x_i) - 0.5}{n}\right) \tag{3}$$

Here, $\text{rank}(x_i)$ is the rank of $x_i$ when the data are sorted in ascending order, $n$ is the number of data points for the consumer in question, and $Q$ is the quantile function derived from the Gaussian distribution. The transformed value $\hat{x_i}$ replaces the original $x_i$ in the scaled dataset. Fig. 3a illustrates the original scaled data across all consumers, while 3b demonstrates the data post-quantile transformation, showcasing the uniformity in scale and distribution achieved across features.

### B. *Data Resampling*

Imbalanced datasets can skew machine learning models, impairing their ability to predict minority class outcomes. To mitigate this, we apply the SMOTE-ENN technique, enhancing the traditional SMOTE algorithm's synthetic sample generation by integrating Edited Nearest Neighbors (ENN) for refinement [42]. This hybrid approach not only augments the minority class with interpolated instances but also prunes samples that could blur the classification boundaries.

For the SGCC dataset, characterized by a class imbalance ratio of $|C_h| : |C_d| = 0.91n : 0.09n$, the SMOTE-ENN process begins by generating synthetic instances $x_{\text{new}}$ from each minority class instance $x$ and its $k$ nearest neighbors $x_{\text{neighbor}}$, using the equation:

$$x_{\text{new}} = x + \lambda \times (x_{\text{neighbor}} - x) \tag{4}$$

where $\lambda$ is a random number between 0 and 1.

Following SMOTE, ENN identifies and removes synthetic instances that are too close to the majority class boundary, based on the majority class neighbor fraction $(p)$:

$$x_{\text{remove}} = \begin{cases} x & \text{if } p > 0.5 \\ \text{retain} & \text{otherwise} \end{cases} \tag{5}$$

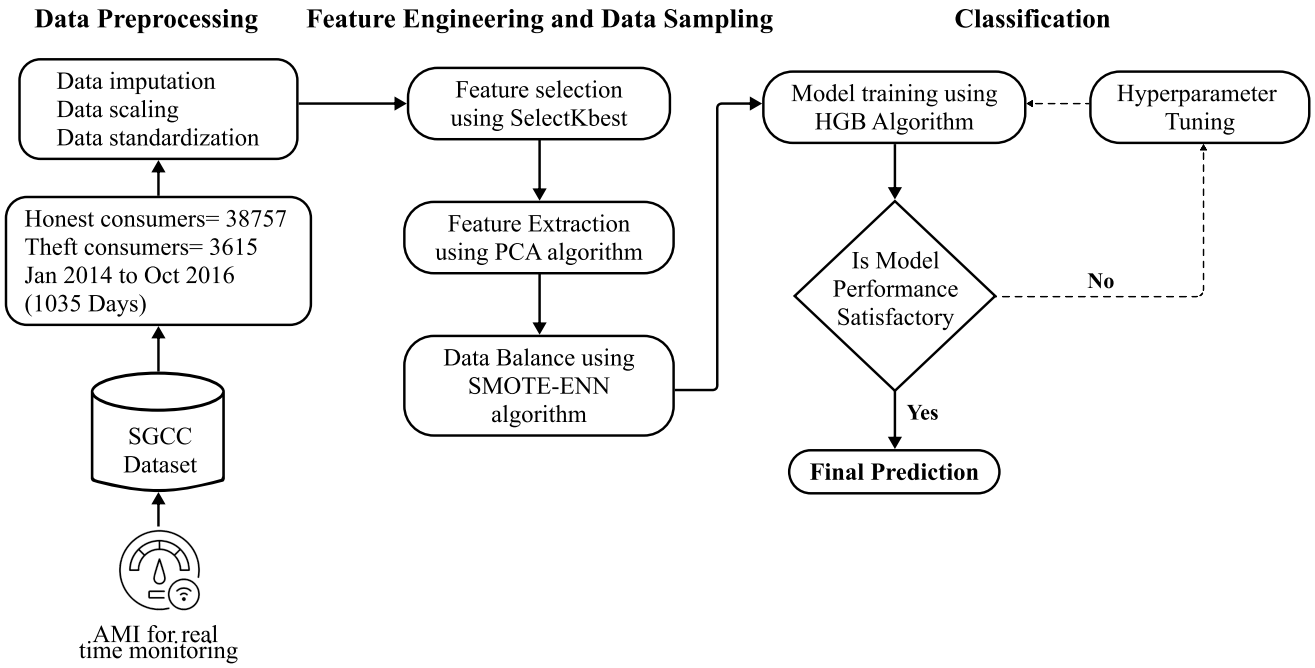


Fig. 2: Process Map of the Proposed Methodology (iHGB)

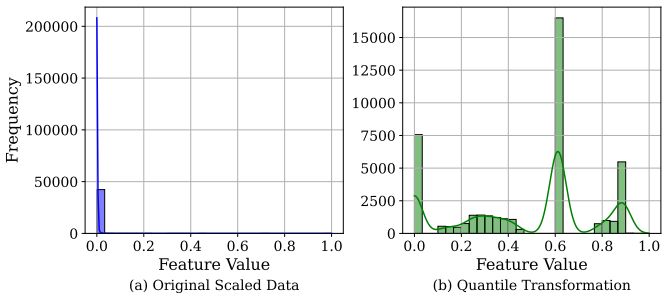


Fig. 3: Feature Distribution before and after QT

The efficacy of the proposed resampling technique and its contribution to the overall accuracy of our model is further discussed in Section III, which considers the empirical results and their implications for the SGCC dataset.

### C. *Feature Engineering*

Feature engineering process plays a critical role in increasing the effectiveness of predictive models. This process of selecting, modifying, and creating features from raw data is vital for disclosing significant insights that might otherwise remain unnoticed [23]. By refining the input data through feature engineering, predictive models can focus on relevant information, resulting in more accurate and reliable predictions. It not only reduces data dimensionality, which reduces computational needs, but also helps in avoiding overfitting to improve model generalizability. In essence, feature engineering is crucial in transforming data into useful information, a critical step that heavily influences the success of the predictive task.

**Algorithm 1** iHGB-Based Approach for NTL Identification

**Input**: Dataset $\mathcal{D}$ with features $\mathbf{x}_i$ and binary labels $y_i$.
**Output**: Optimized HGB Model $\mathcal{M}$.
1: Split $\mathcal{D}$ into $\mathcal{D}_{train}$ and $\mathcal{D}_{test}$ (80% : 20%).
2: Impute missing values in $\mathcal{D}_{train}$ and $\mathcal{D}_{test}$.
3: Standardize $\mathcal{D}_{train}$ and $\mathcal{D}_{test}$ with RS and QT.
4: Feature selection on $\mathcal{D}_{train}$ using SKB for set $\mathcal{F}$.
5: Apply PCA on $\mathcal{D}_{train}$ for dimensionality reduction to $\mathcal{F}_{PCA}$.
6: Balance $\mathcal{D}_{train}$ with SMOTE-ENN for dataset $\mathcal{D}^*_{train}$.
7: Define hyperparameter space for HGB.
8: Perform random search to find the optimal hyperparameters:
- Define the number of iterations $N$ for random search.
- For each iteration, sample a set of hyperparameters.
- Train HGB on $\mathcal{D}^*_{train}$ with current hyperparameters.
- Evaluate performance on a validation set.
- Keep track of the best performing hyperparameters.

9: Train HGB on $\mathcal{D}^*_{train}$ with the best hyperparameters from random search.
10: Test $\mathcal{M}$ on $\mathcal{D}_{test}$ and compare with ADB, GB, LGB.
11: Deploy $\mathcal{M}$ if it outperforms others.

*1) **Feature Selection**:* The SKB algorithm is used to determine the most statistically important features for the binary classification problem of detecting electricity theft [34]. SKB

operates on a univariate basis, evaluating each feature independently using a preset statistical test and maintaining just the top $k$ features that are most relevant to the target variables. The scoring function $f_{classif}$ computes the F-value from the analysis of variance (ANOVA) given the binary nature of given task. This F-value reflects how important the feature is to the predictive model. The F-value for each feature is calculated using the formula:

$$f = (n-2)\frac{r_i^2}{1-r_i^2} \tag{6}$$

where $f$ represents the computed F-value, $n$ is the total number of samples, and $r_i$ is the Pearson correlation coefficient between the $i$-th feature and the target variable. The correlation coefficient $r_i$ is determined by the following equation:

$$r = \frac{\sum(x_i - \overline{x}_i)(y - \overline{y})}{\sqrt{\sum(x_i - \overline{x}_i)^2}\sqrt{\sum(y - \overline{y})^2}} \tag{7}$$

In this context, $x_i$ and $\overline{x}_i$ denote the $i$-th feature vector and its mean, $y$ and $\overline{y}$ represent the target variable and its mean. The selection of $k$ is critical and is typically determined through cross-validation to optimize the model's performance.

*2) **Feature Extraction**:* After performing feature selection, we apply PCA on selected features to reduce the dimensionality of the data [43]. PCA is a statistical method that uses orthogonal transformation to convert a set of possibly correlated observations into a set of linearly uncorrelated values known as principal components. The initial step in PCA is to construct the covariance matrix $C$ from the mean-centred data matrix $X$, which is given by:

$$C = \frac{1}{N-1}X^T X \tag{8}$$

where $N$ is the number of observations in the dataset. The eigen decomposition of the covariance matrix $C$ is performed to obtain the eigenvalues $\lambda$ and the corresponding eigenvectors $v$, which define the principal components. These components are the directions in the feature space along which the data varies the most. Once the principal components are determined, the dataset can be projected onto the subspace spanned by these components. The reduced dataset $Y$ is obtained by:

$$Y = XV_k \tag{9}$$

where $V_k$ is the matrix containing the first $k$ eigenvectors corresponding to the largest $k$ eigenvalues. The number of principal components retained, $k$, is chosen based on the cumulative explained variance, which measures the total variance captured by the first $k$ components. We retain 10 components to ensure that a substantial amount of the original variance is preserved, thereby maintaining the dataset's intrinsic structure, while also reducing the computational load, which is crucial for the scalability and speed of subsequent analyses.

### *D. Classifier Adjustment*

Post two-stage feature selection and extraction procedures, the superfluous and duplicative features have been effectively filtered out. This section describes our proposed algorithm to perform the final classification task on the processed data. We employ the HGB algorithm, which is an advanced variant of traditional GBDT, and is known to be robust and efficient for predictive modelling challenges within large datasets. The algorithm leverages histograms and optimised data structures to improve computational speed and predictive adeptness [44]. In this section, we investigate the formulated classification problem and Bayesian optimization technique to fine-tune HGB hyperparameters.

*1) **Problem Formulation**:* The HGB model employs a sophisticated ensemble of decision trees to make binary predictions with high accuracy. The effect of each tree on the final prediction for a sample $x_i$ is described by the equation:

$$f(x_i;\Theta) = \sum_{k=1}^{K}\theta_k h_k(x_i) \tag{10}$$

where $\theta_k$ denotes the weight, and $h_k(x_i)$ is the corresponding prediction of the $k^{th}$ tree for sample $x_i$. The model's predictive performance is evaluated using the binary cross-entropy loss function, which access the difference between real labels and the predicted probabilities using:

$$L(y_i,\hat{y}_i) = -[y_i\log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] \tag{11}$$

where $y_i$ represents the actual label, and $\hat{y}_i$, derived from $f(x_i;\Theta)$, denotes the predicted probability of the sample belonging to class 1. The primary objective of the HGB model is to minimize this loss function, optimizing the ensemble weights $\Theta$ for better predictive performance. To achieve this with greater computational efficiency, the model adopts a histogram-based strategy for processing continuous input features:

$$X_{hist} = H(x,y) = [H_{ij}] \tag{12}$$

with $H_{ij}$ representing the number of observations for feature $j$ within bin $i$. Throughout the training phase, the HGB model applies a methodical gradient-based optimization technique. At each iteration $t$, it computes the negative gradient of the loss function with respect to the predicted probabilities. This computation directs the construction of a new decision tree:

$$G_{i,t} = -\left(\frac{\partial L(y_i,\hat{y}_i)}{\partial \hat{y}_i}\right) \tag{13}$$

which subsequently identifies the optimal regions in the feature space for the tree $h_t$ to learn from:

$$h_t(x) = \arg\min_R \sum_{i:x_i\in R} G_{i,t} \tag{14}$$

The model updates its predictions by integrating the new tree's output, modulated by a learning rate $\lambda$, into the preceding predictions:

$$\hat{y}_{i,t+1} = \hat{y}_{i,t} + \lambda h_t(x_i) \tag{15}$$

In the final step of the iteration, the ensemble weights $\Theta$ are refined via a gradient descent method to decrease the loss further, adhering to the update formula:

$$\Theta_{t+1} = \Theta_t - \eta\cdot\nabla_\Theta L \tag{16}$$

TABLE I: Optimized HGB Hyperparameters

| Hyperparameter | Tested Range | Optimal Value |
|---|---|---|
| Learning Rate | 0.01 to 0.1 | 0.04 |
| Max Leaf Nodes | 1 to 50 | 3 |
| Max Depth | 1 to 15 | 3 |
| Min Samples Leaf | 1 to 5 | 2 |

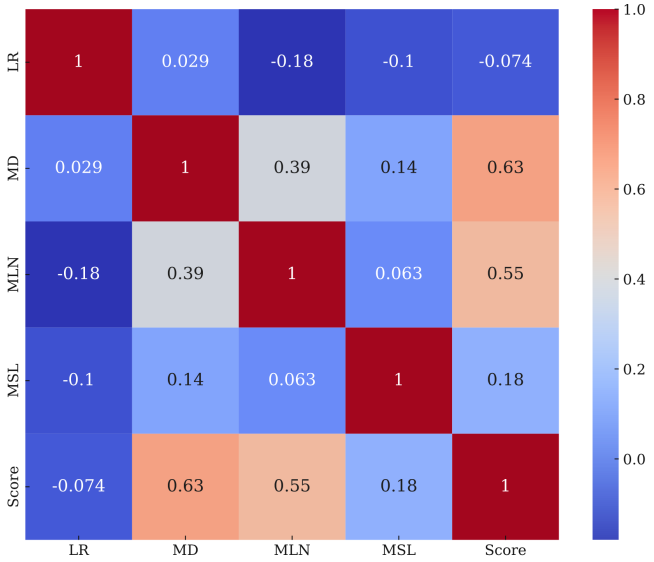


Fig. 4: Hyperparameter Correlation and Model Performance

where $\Theta_{t+1}$ denotes the newly adjusted weights, $\Theta_t$ the weights from the previous iteration, $\eta$ the learning rate of the gradient descent, and $\nabla_\Theta L$ the gradient of the loss function relative to the weights. This iterative enhancement of decision trees and ensemble weights persists until the model reaches an optimal parameter configuration or fulfils a predetermined iteration count.

*2) **Optimal Classification**:* To optimize the performance of the proposed model, Bayesian optimization was applied, targeting an enhancement in the ROC AUC score [45]. The procedure is initiated by establishing a hyperparameter search space that includes key parameters such as learning rate, maximum leaf nodes, maximum depth, and minimum samples per leaf [28]. Employing the Tree-structured Parzen Estimator method, the optimization is executed over 100 iterations, navigating the trade-off between the exploration of new parameter regions and the exploitation of those known to yield positive outcomes. The culmination of this process

TABLE II: Consumer Statistics from SGCC Data Collection

| Dataset Information | Description |
|---|---|
| Data collection time frame | 01-01-2014 to 31-10-2016 |
| Total Consumers | 42,472 |
| Normal Users | 38,757 |
| Abnormal Consumers | 3,615 |
| Missing data samples | 25% |
| Percentage of Normal Users | 91.38% |
| Percentage of Abnormal Consumers | 8.62% |
| Dataset file size | 167 MB |

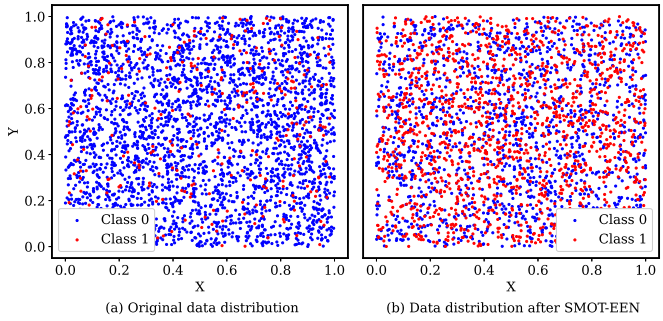


Fig. 5: Data Distribution Before and After Resampling

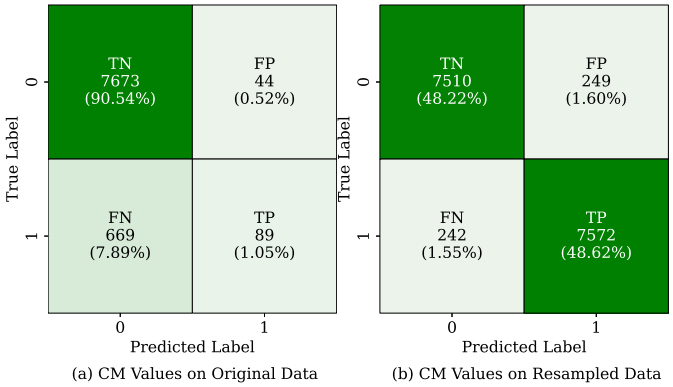


Fig. 6: CM Values Before and After Resampling

is the identification of an optimal set of hyperparameters, subsequently used to train an advanced version of the HGB model on both the untouched and the resampled datasets. This approach not only boosts the model's accuracy but also affords a deeper understanding of hyperparameter influence, aiding in the mitigation of overfitting and improving the model's ability to generalize. The selected hyperparameters, along with their respective ranges and the determined optimal values, are detailed in Table I.

The synergy between hyperparameters and their impact on model performance is depicted in the correlation matrix presented in Fig. 4. The Learning Rate (LR) exhibits a low positive correlation with Max Depth (MD) and minimal negative correlation with Max Leaf Nodes (MLN), suggesting a rather subdued effect on the overall model score. In contrast, MD displays a notable positive correlation with both MLN and the model score, indicating its pivotal role in boosting model performance. Similarly, MLN, despite marginal correlation with Min Samples Leaf (MSL), has a considerable positive influence on the model score, underscoring its importance in the model's predictive success. MLN shows a modest correlation with the score, hence also contributes to the fine-tuning process.

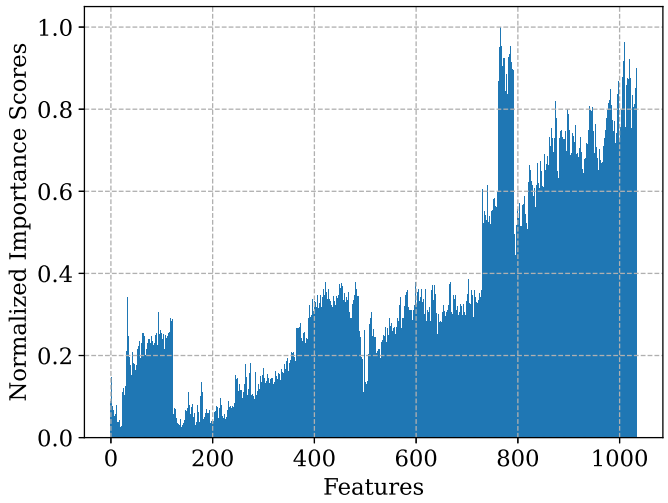


Fig. 7: Importance Score of Each Feature

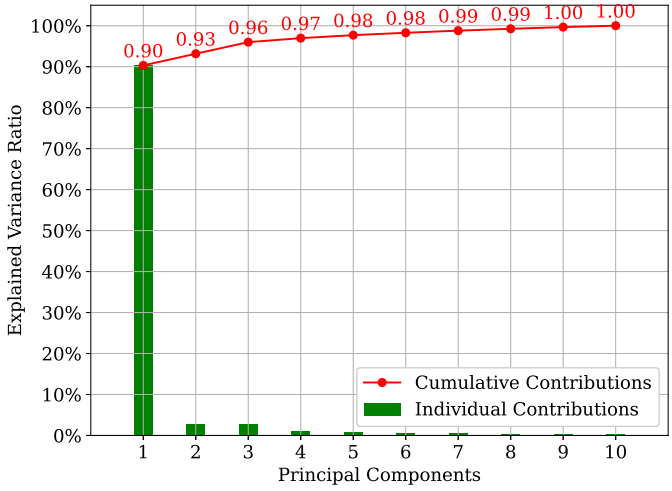


Fig. 8: Cumulative Variance of PCA Components

## III. EXPERIMENTAL SETUP AND RESULTS

### A. *Experimental Setup*

Our analysis utilizes the SGCC's labeled dataset of daily electricity consumption for 42,372 consumers from January 2014 to October 2016, as detailed in Table II. On-site inspections verified consumer classifications, with 38,757 as honest and 3,615 as dishonest. For model validation, we utilized stratified sampling with a random seed to divide the dataset into an 80% training set and a 20% testing set, whilst preserving the original distribution of consumer classifications. This stratification, coupled with the inherent randomness of the sampling, guarantees that our model is trained on a representative and unbiased subset of the data, thereby ensuring a robust evaluation against unseen data. The setup, executed on Google Collaboratory, powered by a MAC i7 processor and 16GB RAM, leveraged a simulator consistent with the system framework described in Section II.

### B. *Performance Results*

*1) **Impact of Data Resampling**:* In this subsection, we compare classification outcomes using two different approaches: without any resampling and with the application of the SMOTE-ENN resampling technique. Figures 5a and 5b demonstrate the distribution of minority and majority classes before and after applying SMOTE-ENN to address imbalanced data. As shown in Fig. 5a, the dominance of the majority class (represented as blue circles) leads to a classifier bias, favoring negative samples. This results in a high True Negative (TN) rate of 90.54%, but also a concerning high False Negative (FN) rate of 7.89% and a low True Positive (TP) rate of 1.55%, as shown via the Confusion Matrix (CM) in Fig. 6a. In the context of ETD, such a high FN rate, which incorrectly classifies fraudulent users as legitimate, poses a significant problem. To mitigate this issue, we employ the SMOTE-ENN technique, which is demonstrated to be effective in Fig. 5b. This method balances the distribution of data, enhancing the model's training and its ability to generalize well on test data. The resultant improvement in model performance, after the application of SMOTE-ENN, is depicted in the CM shown in Fig. 6b. Here, the enhanced balance in data distribution contributes to a more accurate identification of fraudulent behavior, reducing the FN rate and increasing the overall reliability of the ETD system.

*2) **Impact of Feature Engineering**:* Feature engineering plays a pivotal role in improving anomaly detection for electricity theft within the SGCC dataset. We employ the SelectKBest algorithm with ANOVA-F values to systematically prioritize features (i.e. electricity consumption data points), as illustrated in Fig. 7. By setting $k$ = 10, we identify the most influential features, which exhibit a balanced importance distribution: sample 766 (10.5%), 1009 (10.1%), 767 (10.0%), 785 (10.0%), 764 (10.0%), 765 (10.0%), 1010 (10.0%), 784 (9.8%), 763 (9.8%), and 782 (9.8%). This near-uniform distribution (9.8–10.5%) highlights the temporal diversity of theft behaviors in the SGCC dataset. Fraudulent consumers distribute manipulations across different time windows, preventing detection through single temporal patterns. Consequently, multiple time-series features exhibit comparable discriminative power. This observation justifies the application of dimensionality reduction, as these features likely contain subtle temporal correlations that can be effectively captured in a reduced-dimensional space. To address this, PCA was applied, condensing the feature set while retaining essential variance, as demonstrated in Fig. 8. Iterative analysis of different $k$ values reveales a trade-off: while increasing the number of features improves model accuracy, it significantly increases computational complexity. PCA mitigates this trade-off by preserving discriminative information in a more compact representation, enabling efficient processing without compromising predictive performance.

Table III shows the performance evaluation of the proposed algorithm with different preprocessing steps. It shows that feature engineering notably improves the execution time of

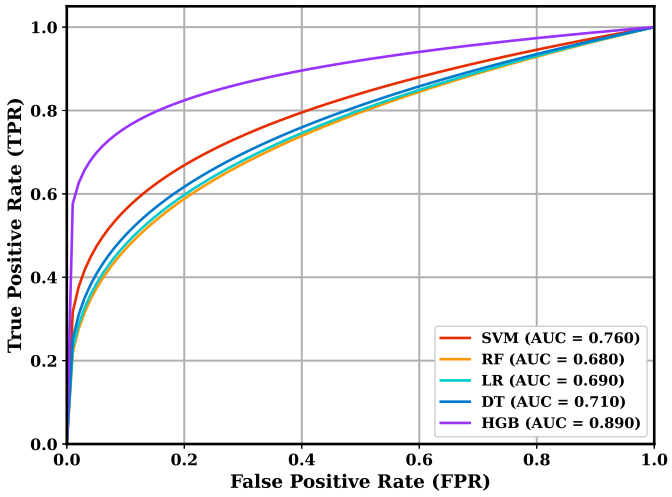


Fig. 9: AUC Comparison with Different Conventional Algorithms

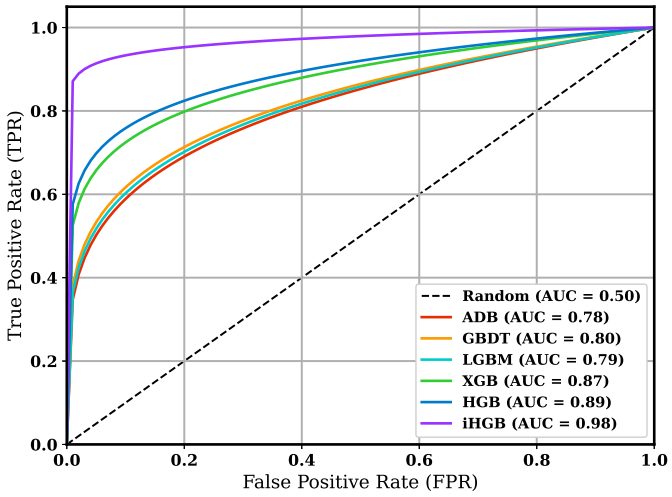


Fig. 10: AUC for Different Variants of Boosting Algorithms

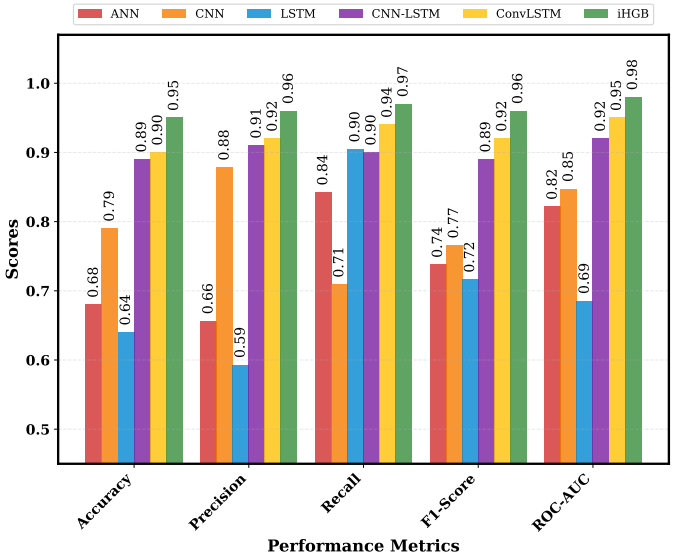


Fig. 11: Performance Comparison of Deep Learning Algorithms

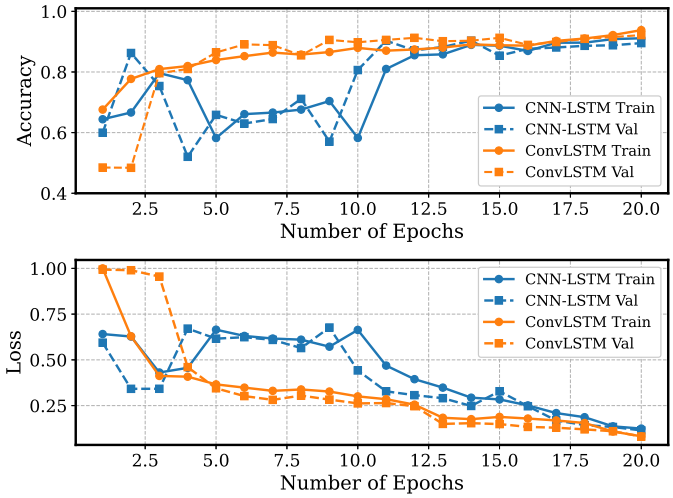


Fig. 12: Hybrid Model Learning Curves Comparison

Framework C by 25% compared to Framework B, albeit with a marginal decrease in accuracy (Frameworks A–E are discussed later). This trade-off underscores the balance required in machine learning between computational efficiency and predictive integrity, especially in real-time anomaly detection scenarios.

*3) **Comparison Among Benchmark Algorithms**:* We compare HGB and the proposed iHGB performance with various machine learning classifiers, including SVM, RF, LR and DT, together with three other well performing ensemble classifiers, using the key performance metrics of accuracy, F1-score, AUC, and execution time. Figs. 9 and 10 illustrate the results for machine learning algorithms and ensemble strategies, respectively. Table III illustrates a detailed performance comparison, where iHGB emerges with the lead across all the performance metrics. Notably, iHGB's accuracy peaks at 97%, coupled with an F1-score of 96%, a testament to its precision and recall balance. Moreover, an AUC of 98% underscores its discriminative power in segregating between both classes.

These substantial gains in performance do not come at the expense of efficiency, as evidenced by a reasonable execution time of 40 seconds. Given that the iHGB model takes 40 seconds to complete training on the 12,458 samples in our training dataset, the per-sample execution time is approximately 0.0032 seconds per sample. This is particularly notable when compared to traditional models such as SVM and DT, which, despite lower performance metrics, exhibit comparable execution times of 77 and 31 seconds, respectively. The substantial margin by which iHGB outperforms its counterparts, ADB, XGB [4], GBM, and LGB, reinforces the impact of Bayesian Optimization on model efficacy. This optimization approach has fine-tuned iHGB to a level of performance that

TABLE III: Performance Evaluation of the Proposed Algorithm with Different Preprocessing Steps

| Techniques | Framework | Accuracy | F1-Score | AUC | Time (Sec) |
|---|---|---|---|---|---|
| Original Data + HGB | A | 0.522 | 0.532 | 0.591 | 18 |
| Data Preprocessing (DP) + HGB | B | 0.792 | 0.740 | 0.821 | 20 |
| DP + Feature Engineering (FE) + HGB | C | 0.701 | 0.739 | 0.754 | 15 |
| DP + FE + Data Balancing (DB) + HGB | D | 0.881 | 0.820 | 0.892 | 44 |
| DP + FE + DB + iHGB | E | 0.970 | 0.962 | 0.981 | 93 |

TABLE IV: Performance Comparison of Different Algorithms

| Algorithms | Accuracy | F1-Score | AUC | Time (Sec) |
|---|---|---|---|---|
| LGB [32] | 0.64 | 0.78 | 0.80 | 7 |
| SVM [36] | 0.67 | 0.66 | 0.78 | 77 |
| LR | 0.62 | 0.69 | 0.76 | 30 |
| DT | 0.68 | 0.72 | 0.79 | 31 |
| RF | 0.65 | 0.69 | 0.77 | 14 |
| ADB | 0.71 | 0.78 | 0.79 | 06 |
| GBM | 0.74 | 0.80 | 0.80 | 32 |
| XGB [4] | 0.81 | 0.84 | 0.87 | 17 |
| HGB | 0.82 | 0.87 | 0.89 | 13 |
| iHGB | 0.95 | 0.96 | 0.98 | 40 |
| ANN | 0.68 | 0.74 | 0.82 | 157 |
| LSTM | 0.64 | 0.72 | 0.71 | 322 |
| CNN | 0.79 | 0.77 | 0.85 | 290 |
| CNN-LSTM | 0.89 | 0.89 | 0.92 | 262 |
| ConvLSTM [8] | 0.90 | 0.92 | 0.95 | 369 |

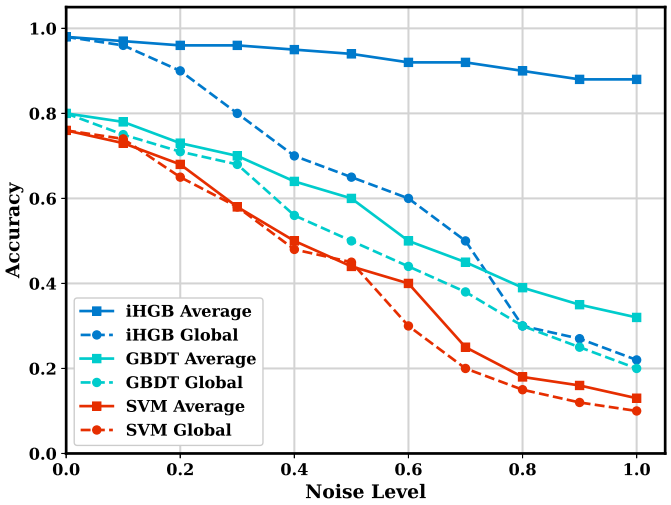


Fig. 13: Robustness Comparison for iHGB and Benchmark Frameworks to Data Noise Conditions

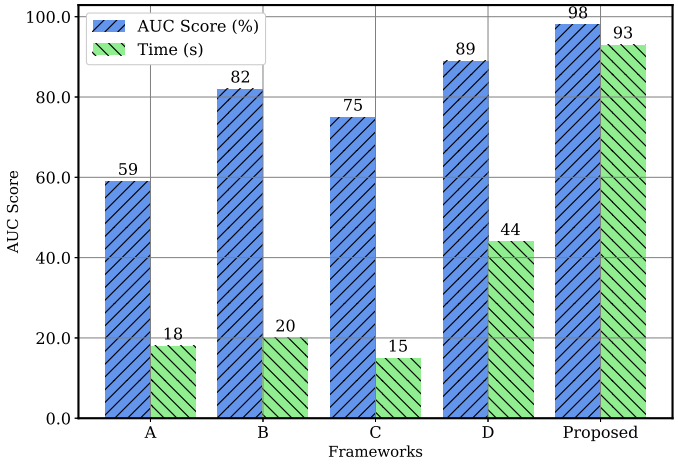


Fig. 14: Improved Proposed Model

arguably sets a new benchmark within the domain, particularly in the context of electrical consumption pattern analysis.

*4) **iHGB Performance Comparison with Deep Learning Models**:* In this empirical investigation, we present a rigorous comparative analysis of various deep learning architectures, and our proposed iHGB model. The evaluation metrics include accuracy, F1-score, AUC, and computational efficiency, as detailed in Table IV and visualized in Fig. 11

The experimental results demonstrate the hierarchical performance patterns across different architectural paradigms. Hybrid architectures, specifically CNN-LSTM and ConvLSTM [8], exhibit superior predictive capabilities compared to standalone models, achieving accuracy rates of 89% and 90% respectively. These architectures leverage both spatial and temporal feature extraction mechanisms, resulting in enhanced discriminative power. As illustrated in Fig. 12, the learning curves for both hybrid models reveal significant convergence limitations. After 20 epochs, performance plateaus at approximately 92% accuracy, demonstrating clear difficulty achieving optimal convergence - a threshold that our proposed iHGB model successfully surpasses. The CNN-LSTM architecture exhibits particularly inconsistent behavior with notable performance fluctuations, while the ConvLSTM shows more stable but ultimately constrained learning progression. The conventional deep learning models show varying degrees of effectiveness: CNN achieves 79% accuracy with an AUC of 0.85, while LSTM and ANN demonstrate more modest performance (64% and 68% accuracy respectively). This performance disparity, clearly visible in the comparative analysis shown in Fig. 11, highlights the limitations of single-architecture approaches in capturing complex data patterns.

Notably, our proposed iHGB model demonstrates improved performance metrics, achieving 95% accuracy and 0.98 AUC score, surpassing the predictive power of hybrid architectures. The superior performance of iHGB is particularly significant given that hybrid models, despite their architectural sophistication and extended training periods, still fall short of matching iHGB's predictive capabilities, as evidenced by the performance metrics visualized in Fig. 11. A critical advantage

TABLE V: Performance of Classifiers at Different Training Ratios

| Classifiers | Training Ratio = 50% | | | Training Ratio = 60% | | | Training Ratio = 70% | | |
|---|---|---|---|---|---|---|---|---|---|
| | Accuracy | F1-Score | AUC | Accuracy | F1-Score | AUC | Accuracy | F1-Score | AUC |
| RF | 0.64 | 0.78 | 0.68 | 0.64 | 0.78 | 0.63 | 0.64 | 0.78 | 0.67 |
| DT | 0.68 | 0.79 | 0.72 | 0.68 | 0.79 | 0.71 | 0.68 | 0.79 | 0.71 |
| LR | 0.64 | 0.76 | 0.69 | 0.64 | 0.76 | 0.68 | 0.64 | 0.76 | 0.68 |
| SVM | 0.71 | 0.79 | 0.75 | 0.71 | 0.79 | 0.75 | 0.72 | 0.79 | 0.75 |
| XGB [4] | 0.74 | 0.76 | 0.78 | 0.78 | 0.79 | 0.80 | 0.80 | 0.82 | 0.83 |
| HGB | 0.79 | 0.84 | 0.86 | 0.79 | 0.85 | 0.87 | 0.80 | 0.85 | 0.87 |
| ConvLSTM [8] | 0.78 | 0.77 | 0.80 | 0.85 | 0.85 | 0.84 | 0.89 | 0.90 | 0.90 |
| iHGB | 0.87 | 0.90 | 0.90 | 0.88 | 0.92 | 0.93 | 0.94 | 0.94 | 0.95 |

of iHGB lies in its computational efficiency. While hybrid models require substantial computational resources (CNN-LSTM: 262s, ConvLSTM: 369s), iHGB achieves superior performance in merely 40 seconds, as demonstrated in Table IV. This optimization in computational overhead, coupled with enhanced predictive accuracy, positions iHGB as an ideal solution for real-time applications requiring both precision and efficiency.

For comprehensive architectural details of hybrid models and their implementations, readers are directed to Reference [8]. The empirical evidence suggests that while hybrid deep learning models offer robust performance, iHGB provides a more efficient and accurate alternative, making it particularly suitable for practical applications demanding both computational efficiency and high predictive accuracy.

*5) **iHGB Robustness Compared with Benchmark Algorithms**:* The robustness of the iHGB model is compared with two of the benchmark algorithms, namely GBDT and SVM. The evaluation was conducted using two different noise introduction methods. Firstly, random noise is added to each feature individually to simulate errors typically caused by malfunctioning sensors. The average accuracy of each algorithm under these conditions is depicted by solid traces in Fig. 13. Secondly, random noise was added globally to selected data points across multiple features to mimic transmission errors in critical environments, with the results shown by dashed traces in Fig. 13. These results suggest that the iHGB model could exhibit superior robustness compared to the benchmark algorithms. The impact of feature-specific noise on the iHGB model's accuracy is minimal, indicating that the model effectively filters out less important features during the selection and extraction process. Furthermore, the iHGB framework shows enhanced resilience to global noise, maintaining high accuracy even when multiple features are affected. This robustness can be attributed to the tuned hyperparameters and the model's ability to mitigate the influence of noisy features, validating the iHGB model's effectiveness for real-world applications where data integrity may be compromised.

*6) **iHGB Scalable Theft Detection Comparative Performance Analysis**:* To ascertain the efficacy of the iHGB algorithm, we performed the comparative analyses summarized in Table IV and Fig. 14. These help to corroborate iHGB's enhanced accuracy in detecting theft over the benchmarks. This comparative evaluation across frameworks A through D and our proposed system E confirms that each integrated module in our design contributes to the heightened accuracy of theft detection. The iHGB algorithm, through the reduction of irrelevant and redundant features, coupled with the fine-tuning of hyperparameters via Bayesian Optimization, ensures elevated accuracy in identifying instances of electricity theft.

Finally, we explore the scalability of our framework by varying the sizes of the training subsets and monitoring the resultant average accuracy, as tabulated in Table V. The iHGB algorithm consistently outshines the benchmarks, achieving AUC scores of 94.2%, 95.1%, and 95.7%, with training subsets of 50%, 60%, and 70%, respectively, thus surpassing the standard HGB and other models. Our framework demonstrates greater scalability in comparison to the five benchmarks. Notably, the size of the training subset exerts minimal influence on the accuracy of our model, attributable to the discarding of non-essential features during the feature selection and extraction phases, enhancing the model's efficiency and predictive performance.

## IV. CONCLUSIONS

This article has presented a robust NTL detection framework, anchored by a novel application of HGB. Data deficiencies and imbalances are addressed first, thereby laying the groundwork for precise data analysis. The application of SMOTE-ENN effectively normalizes data distribution, while SKB and PCA refine the feature selection process, enhancing the classifier's performance and efficiency. Central to the study's success is the deployment of the HGB algorithm, which differentiates between legitimate and fraudulent energy use. This system's precision is improved by Bayesian optimization, which fine-tunes the model to prevent overfitting and ensure broader applicability. Benchmarked against contemporary models, the new framework surpasses standard performance metrics, achieving 93.4% accuracy, a 96% F1 score, and a 98% AUC.

In future research, we aim to assess the potential of unsupervised methodologies, with an emphasis on deep learning techniques, to enhance the precision of NTL detection. The motivation behind this direction is the prospect of uncovering latent correlations within the dataset that might be elusive to traditional approaches.

## ACKNOWLEDGMENT

This work is partially supported by the Leverhulme Trust under grant number RPG-2023-107.

## REFERENCES

[1] A. O. Otuoze, M. W. Mustafa, O. O. Mohammed, M. S. Saeed, N. T. Surajudeen-Bakinde, and S. Salisu, "Electricity theft detection by sources of threats for smart city planning," IET Smart Cities, pp. 52–60, 2019.

[2] Yan, Zhongzong, and He Wen. "Performance analysis of electricity theft detection for the smart grid: An overview." IEEE Transactions on Instrumentation and Measurement 71 (2021): 1-28.

[3] I. U. Khan, N. Javeid, C. J. Taylor, K. A. A. Gamage and X. Ma, "A Stacked Machine and Deep Learning-Based Approach for Analysing Electricity Theft in Smart Grids," in IEEE Transactions on Smart Grid, vol. 13, no. 2, pp. 1633-1644, March 2022.

[4] Z. Yan and H. Wen, "Electricity theft detection base on extreme gradient boosting in ami," IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1–9, 2021.

[5] Yao, Ruizhe, Ning Wang, Weipeng Ke, Zhili Liu, Zhenhong Yan, and Xianjun Sheng. "Electricity Theft Detection in Incremental Scenario: A Novel Semi-supervised Approach based on Hybrid Replay Strategy." IEEE Transactions on Instrumentation and Measurement (2023).

[6] Qi, Ruobin, Jun Zheng, Zhirui Luo, and Qingqing Li. "A novel unsupervised data-driven method for electricity theft detection in AMI using observer meters." IEEE Transactions on Instrumentation and Measurement 71 (2022): 1-10.

[7] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed, and M. Mohamad, "Nontechnical loss detection for metered customers in power utility using support vector machines," IEEE Transactions on Power Delivery, vol. 25, no. 2, pp. 1162–1171, 2010.

[8] Gao, Hong-Xin, Stefanie Kuenzel, and Xiao-Yu Zhang. "A hybrid ConvLSTM-based anomaly detection approach for combating energy theft." IEEE Transactions on Instrumentation and Measurement 71 (2022): 1-10.

[9] K. Zheng, Q. Chen, Y. Wang, C. Kang, and Q. Xia, "A novel combined data-driven approach for electricity theft detection," IEEE Transactions on Industrial Informatics, vol. 15, no. 3, pp. 1809–1819, 2019.

[10] Yao, R., Wang, N., Ke, W., Liu, Z., Yan, Z., & Sheng, X. (2023). Electricity Theft Detection in Incremental Scenario: A Novel Semi-Supervised Approach Based on Hybrid Replay Strategy. IEEE Transactions on Instrumentation and Measurement, 72. https://doi.org/10.1109/TIM.2023.3324674

[11] Y. Guo, C.-W. Ten, and P. Jirutitijaroen, "Online data validation for distribution operations against cybertampering," IEEE Transactions on Power Systems, vol. 29, no. 2, pp. 550–560, 2013.

[12] F. G. Guarda, B. K. Hammerschmitt, M. B. Capeletti, N. K. Neto, L. L. dos Santos, L. R. Prade, and A. Abaide, "Non-hardware-based non-technical losses detection methods: A review," Energies, vol. 16, no. 4, p. 2054, 2023.

[13] N. Javaid, U. Qasim, A. S. Yahaya, E. H. Alkhammash, M. Hadjouni et al., "Non-technical losses detection using autoencoder and bidirectional gated recurrent unit to secure smart grids," IEEE Access, vol. 10, pp. 56 863–56 875, 2022.

[14] R. Morales-Caporal, "Modular Advanced Metering Infrastructure to Reduce Electricity Theft and a Cluster-Based Illegal Loads Detection," in IEEE Latin America Transactions, vol. 21, no. 4, pp. 579-587, April 2023, doi: 10.1109/TLA.2023.10128930.

[15] C.-H. Lo and N. Ansari, "Consumer: A novel hybrid intrusion detection system for distribution networks in smart grid," IEEE Transactions on Emerging Topics in Computing, vol. 1, no. 1, pp. 33–44, 2013.

[16] L. A. P. Junior, C. C. O. Ramos, D. Rodrigues, D. R. Pereira, A. N. de Souza, K. A. P. da Costa, and J. P. Papa, "Unsupervised non-technical losses identification through optimum-path forest," Electric Power Systems Research, vol. 140, pp. 413–423, 2016.

[17] M. N. Hasan, R. N. Toma, A.-A. Nahid, M. M. Islam, and J.-M. Kim, "Electricity theft detection in smart grid systems: A cnn-lstm based approach," Energies, vol. 12, no. 17, p. 3310, 2019.

[18] Pamir, N. Javaid, U. Qasim, A. S. Yahaya, E. H. Alkhammash, and M. Hadjouni, "Non-technical losses detection using autoencoder and bidirectional gated recurrent unit to secure smart grids," IEEE Access, vol. 10, pp. 56 863–56 875, 2022.

[19] Z. Zheng, Y. Yang, X. Niu, H.-N. Dai, and Y. Zhou, "Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids," IEEE Transactions on Industrial Informatics, vol. 14, no. 4, pp. 1606–1615, 2017.

[20] P. Jokar, N. Arianpoo, and V. C. Leung, "Electricity theft detection in ami using customers' consumption patterns," IEEE Transactions on Smart Grid, vol. 7, no. 1, pp. 216–226, 2015.

[21] L. Cui, L. Guo, L. Gao, B. Cai, Y. Qu, Y. Zhou, and S. Yu, "A covert electricity-theft cyberattack against machine learning-based detection models," IEEE Transactions on Industrial Informatics, vol. 18, no. 11, pp. 7824–7833, 2022.

[22] A. H. Nizar, Z. Y. Dong, and P. Zhang, "Detection rules for non technical losses analysis in power utilities," in 2008 IEEE Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century. IEEE, 2008, pp. 1–8.

[23] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," Computers & Electrical Engineering, vol. 40, no. 1, pp. 16–28, 2014.

[24] R. Razavi, A. Gharipour, M. Fleury, and I. J. Akpan, "A practical feature-engineering framework for electricity theft detection in smart grids," Applied energy, vol. 238, pp. 481–494, 2019.

[25] Tureczek, Alexander Martin, and Per Sieverts Nielsen. "Structured literature review of electricity consumption classification using smart meter data." Energies 10.5 (2017).

[26] N. Javaid, N. Jan, and M. U. Javed, "An adaptive synthesis to handle imbalanced big data with deep siamese network for electricity theft detection in smart grids," Journal of Parallel and Distributed Computing, vol. 153, pp. 44–52, 2021.

[27] Adil, Muhammad, et al. "LSTM and bat-based RUSBoost approach for electricity theft detection." Applied Sciences 10.12 (2020): 4378.

[28] I. U. Khan, N. Javaid, C. J. Taylor and X. Ma, "Robust Data Driven Analysis for Electricity Theft Attack-Resilient Power Grid," in IEEE Transactions on Power Systems, vol. 38, no. 1, pp. 537-548, Jan. 2023.

[29] Gunturi, Sravan Kumar, and Dipu Sarkar. "Ensemble machine learning models for the detection of energy theft." Electric Power Systems Research 192 (2021)

[30] Buzau, Madalina Mihaela, et al. "Detection of non-technical losses using smart meter data and supervised learning." IEEE Transactions on Smart Grid 10.3 (2018): 2661-2670.

[31] Oprea, Simona-Vasilica, and Adela Bâra. "Feature engineering solution with structured query language analytic functions in detecting electricity frauds using machine learning." Scientific Reports (2022).

[32] Hussain, Saddam, Mohd Wazir Mustafa, Touqeer A. Jumani, Shadi Khan Baloch, Hammad Alotaibi, Ilyas Khan, and Afrasyab Khan. "A novel feature engineered-CatBoost-based supervised machine learning framework for electricity theft detection." Energy Reports 7 (2021).

[33] Upadhyay, Darshana, et al. "Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids." IEEE Transactions on Network and Service Management 18.1 (2020).

[34] Munawar, Shoaib, et al. "Electricity Theft Detection in Smart Grids Using a Hybrid BiGRU–BiLSTM Model with Feature Engineering-Based Preprocessing." Sensors 22.20 (2022): 7818.

[35] Javaid, Nadeem, et al. "Non-technical losses detection using autoencoder and bidirectional gated recurrent unit to secure smart grids." IEEE Access.

[36] Shehzad, Faisal, Nadeem Javaid, Sheraz Aslam, and Muhammad Umar Javed. "Electricity theft detection using big data and genetic algorithm in electric power systems." Electric Power Systems Research 209 (2022).

[37] R. Punmiya and S. Choe, "Energy theft detection using gradient boosting theft detector with feature engineering-based preprocessing," IEEE Transactions on Smart Grid, vol. 10, no. 2, pp. 2326–2329, 2019.

[38] H.-X. Gao, S. Kuenzel, and X.-Y. Zhang, "A hybrid convlstm- based anomaly detection approach for combating energy theft," IEEE Transactions on Instrumentation and Measurement, vol. 71, pp. 1–10, 2022.

[39] A. Takiddin, M. Ismail, M. Nabil, M. M. E. A. Mahmoud, and E. Serpedin, "Detecting electricity theft cyber-attacks in ami networks using deep vector embeddings," IEEE Systems Journal, vol. 15, no. 3, pp. 4189–4198, 2021.

[40] D. Micci-Barreca, "A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems," ACM SIGKDD Explorations Newsletter, vol. 3, no. 1, pp. 27–32, 2001.

[41] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, "Study the influence of normalization/transformation process on the accuracy of supervised classification," in 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020, pp. 729–735.

[42] D. Elreedy and A. F. Atiya, "A comprehensive analysis of synthetic minority oversampling technique (smote) for handling class imbal- ance," Information Sciences, vol. 505, pp. 32–64, 2019.
[43] J. Brownlee, "Feature selection in python with selectkbest," https://machinelearningmastery.com/ feature-selection-with-real-and-categorical-data/, 2020, accessed: 25, 6, 2023.
[44] J. Cui, H. Hang, Y. Wang, and Z. Lin, "Gbht: Gradient boosting histogram transform for density estimation," in International Conference on Machine Learning. PMLR, 2021, pp. 2233–2243.
[45] Pelikan, M., & Pelikan, M. (2005). Bayesian optimization algorithm. Hierarchical Bayesian optimization algorithm: toward a new generation of evolutionary algorithms, 31-48.