Hybrid CNN-LSTM Based Anomaly Detection for Energy Theft in Residential Smart Meter Data

1st Hasina Rahman
School of Data and Electrical Engineering
University of Technology Sydney
Sydney, Australia
hasina.rahman@student.uts.edu.au

2nd Nazim Uddin Sheikh
Business Information Systems
AIH
Sydney, Australia
n.sheikh@aih.edu.au

3rd Priyadarsi Nanda
School of Data and Electrical Engineering
University of Technology Sydney
Sydney, Australia
priyadarsi.nanda@uts.edu.au

***Abstract*—Energy theft is a critical issue plaguing modern smart grids, leading to significant losses, distorted demand profiles, and compromised grid stability. The smart meter data can be tampered manually as well as through cyber-attacks. These disruptions not only undermine the financial sustainability of power utilities but also burden honest consumers through higher tariffs and unreliable service. To address this challenge, we propose a scalable anomaly-based theft detection framework using a CNN-LSTM autoencoder. Our scheme uses an unsupervised machine learning technique and resource constraint IoT friendly lightweight model. The approach leverages temporal patterns in smart meter data, enabling accurate reconstruction and identification of anomalous consumption behavior related to theft. We enhance robustness by incorporating cluster-specific post-processing techniques—combining reconstruction errors with multiple statistical anomaly detectors. Our scheme is based on a hybrid model and voting strategy that flags anomalies only when multiple detectors concur, significantly reducing false positives. Furthermore, domain-specific theft detection rules are applied based on sudden spikes, sustained high usage, and unusual night-time consumption within each behavioral cluster. Experimental results using real-world smart meter datasets demonstrate the effectiveness of our model in isolating theft instances with high precision rate of 100% for one of our clusters while maintaining generalization across diverse consumer profiles. The proposed system supports proactive monitoring and can assist utility providers in minimizing non-technical losses while preserving trust and equity in energy distribution.**

***Index Terms*—Smart Metering, Anomaly Detection, CNN-LSTM, Energy Theft, Clustering, Autoencoder**

I. Introduction

The amount of energy loss during the process of generation, transmission and distribution of electricity is incomprehensible. These huge losses can broadly be categorized into technical and non-technical losses [1]. Initially, more than 60% of the energy used is lost in conversion as mentioned in the 2019 U.S. Energy Information Administration (EIA)[1] report that falls under the category of technical loss. However, in general, the technical losses are considered to be caused due to energy dissipation in conductors, transformers, and other equipment during transmission and distribution. On the other hand, non-technical losses are categorized as commercial losses which caused by energy theft, faulty meters, billing errors, insider fraud and lack of real-time monitoring [2]. These issues can be addressed using the advanced technology in smart grids that reduces reliance on inefficient peak generators through demand forecasting and distributed energy resource (DER) integration. Intelligent dispatch, storage, and real-time controls minimize auxiliary and start-stop losses in generation [3]. Also, smart grids enhance network efficiency through real-time load balancing, voltage optimization, and dynamic line rating (DLR). Automated fault detection and predictive maintenance further reduce resistive and equipment losses [4]. Additionally, Advanced Metering Infrastructure (AMI) supports real-time monitoring, remote disconnection, and tamper alerts. Further, data analytics facilitated through the availability of huge smart meter usage data can help detect anomalous consumption suggesting theft, while automated billing reduces human error and fraud [5]. However, it also paves way for new attack methods that may lead to energy thefts or other cyber attacks.

Globally, energy theft accounts for substantial financial losses. According to the World Bank, non-technical losses including theft can exceed $96 billion annually [6]. As already mentioned, countries such as India and Brazil, estimates suggest that the theft rates as high as 20–30% of distributed electricity in certain regions [7] The energy losses adversely affect grid reliability, inflate electricity tariffs for legitimate users, and discourage investment in this sector. Furthermore, unauthorized meter tampering and illegal connections introduce critical safety hazards, often resulting in electrocution, fires, and overloaded distribution networks [8].

More recent advances combine multiple model types. For instance, CNN-LSTM hybrids have been used to exploit both spatial and temporal aspects of the consumption data [9]. Further, transformer-based architectures have demonstrated even higher accuracy due to their attention mechanisms [10]. However, many of these models operate in Euclidean domains and do not natively handle graph-structured data, which is crucial when considering topological or relational aspects among meters. Graph Convolutional Networks (GCNs) address this limitation by leveraging graph representations, capturing complex correlations [11] and enabling effective node-level anomaly detection [12].

In this paper, we present a cluster-aware unsupervised CNN-LSTM autoencoder framework for energy theft detection. Unlike previous works that relies solely on either Euclidean CNNs

[1]https://www.eia.gov/todayinenergy/detail.php?id=44436

or GCNs, the proposed framework integrates both behavioral and anomaly detection paradigms to balance interpretability, performance, and deployment scalability [13].

Our Contribution:

We summarise our contributions as follows.

- **Cluster-Aware Detection:** We augment input data with behavioral cluster labels, learned via KMeans from daily profile patterns. This allows the model to detect anomalies relative to behavioral baselines, rather than applying uniform global thresholds.
- **Scalable:** Since our scheme classifies meters into clusters, one global model is trained using meters with different consumption behaviors as labels. This enables training on selected meters from different clusters rather than large number of meters and maintaining separate models.
- **Temporal Modeling:** We propose a hybrid CNN-LSTM autoencoder to model both spatial (local temporal) and long-range sequential dependencies, enhancing anomaly detection capabilities without relying on labels.
- **Lightweight and edge deployable:** Our model is lightweight and it's significance emerges from the quality of being deployed on any smart-meter edge device.
- **Ensemble Anomaly Detection and Rule-Based Theft Inference:** Our detection framework combines statistical detectors (e.g., Z-score, EMA, Isolation Forest) with interpretable theft rules tailored to specific consumption clusters. This hybrid design balances detection performance, interpretability, and deployment scalability.

II. Related Works

Electricity theft detection in smart grids has been widely explored through machine learning and deep learning techniques. Traditional statistical and rule-based methods often suffer from limited generalizability and low robustness, particularly when user behavior patterns are diverse or when theft techniques evolve. Consequently, data-driven approaches—especially deep learning—have gained traction due to their ability to model nonlinear and temporal patterns in consumption data.

Zheng et al. [14] introduced a Wide and Deep Convolutional Neural Network (WDCNN) for theft detection, integrating wide shallow layers and deep CNN modules to capture both global and local patterns. Although effective at modeling multiscale spatial features, their method lacks explicit temporal modeling, which is crucial for sequential anomaly detection in smart meter data.Mangat et al. [15] proposed a multi-layer deep neural network (DNN) classifier using handcrafted features derived from smart meter readings. While this supervised approach achieved good accuracy on benchmark datasets, it relied heavily on feature engineering and did not generalize well to unseen or evolving consumption behaviors. To incorporate spatial structure, Liao et al. [1] employed a Graph Convolutional Network (GCN) for theft detection, capturing relationships between users based on grid topology and spatial correlations. However, GCN models often require graph construction based on physical infrastructure, which may not always be feasible or available in anonymized or large-scale datasets. Feng et al. [16] proposed using Text-CNNs by encoding consumption sequences as symbolic representations. Although innovative, this transformation introduces abstraction overhead and may result in loss of granular temporal detail that is often critical for identifying subtle theft patterns. Rouzbahani et al. [17] explored an ensemble framework combining multiple CNN architectures to boost detection robustness. While ensemble methods offer improved generalization, they tend to be computationally intensive, limiting their practicality in real-time or edge-deployed scenarios such as smart meters and gateways.

While previous works have explored either spatial or statistical aspects of energy theft, our approach differs significantly as mentioned in the contribution section. Our work is therefore well-aligned with practical deployment constraints, including the use of unlabeled data, the need for lightweight models, and real-time responsiveness on edge devices.

III. Methodology

In this section, we put forward an end-to-end unsupervised framework for electricity theft detection. Our proposed framework is based on five core components which combine temporal clustering, CNN-LSTM auto-encoding, ensemble anomaly detection, and rule-based theft inference. The pipeline consists of the following components: temporal behavioural clustering; data processing & sequence construction; global and lightweight CNN-LSTM autoencoder; ensemble based anomaly detection; rule-based theft detection. We explain each component as follows.

A. Temporal Behavioural Clustering using KMeans

A profile matrix was constructed by aggregating normalized daily profiles across days for each meter. Clustering was then performed using KMeans clustering algorithm, with the optimal number of clusters determined via the elbow method.

Each meter was assigned a behaviour cluster label (one-hot encoded) for later augmentation into model inputs.

B. Data Preprocessing and Sequence Construction

We engineered multiple datetime and features extracted from the energy consumption records (e.g., daily sum, max, min, monthly mean) for each meter. The features were normalized using Box-Cox transformation [18] and scaled with StandardScaler. Temporal sequences were generated with a lookback window of 48 time steps (24 hours per day). The cluster labels were appended to each sequence to add behavioral context.

C. Global CNN-LSTM Autoencoder

We trained a shallow CNN-LSTM autoencoder on the aggregated sequences from all selected meters. The model is designed to be lightweight for the ease of deployment on edge devices in order to preserve privacy. This would help in complying with GDPR for smart meter data. Additionally, it is essentially unsupervised to deal with real-world data since data is rarely labeled.

*1) **CNN-LSTM Autoencoder**:* We have configured a hybrid model based on convolutional neural network and long short-term memory (CNN-LSTM) autoencoder to learn typical consumption behavior and detect anomalies in smart meter data. The model combines spatial feature extraction (via CNN) and temporal sequence modeling (via LSTM) to leverage both short-term and long-term consumption patterns [19]–[22].

Our deep learning model architecture includes:

- 1D convolution and max-pooling to extract local features;
- LSTM layers to learn long-term dependencies; and
- a decoder to reconstruct the input sequence.

TABLE I: CNN-BiLSTM Autoencoder Layer Description

Layer	Function and Purpose
Conv1D(64)	Extracts local temporal patterns using 64 filters (ReLU, kernel size = 3)
BatchNorm	Stabilizes training by normalizing activations
MaxPooling1D(2)	Downsamples time dimension by factor of 2
Dropout(0.2)	Regularizes model by dropping 20% of neurons
Conv1D(32)	Learns deeper temporal features with fewer filters
BatchNorm	Further normalization of feature maps
MaxPooling1D(2)	Further reduces temporal resolution
Dropout(0.2)	Regularization at deeper layers
Bi-LSTM(64)	Captures long-term and bidirectional dependencies
UpSampling1D(2)	Restores original time resolution
Conv1D(32)	Refines upsampled features
TimeDistributed(Dense)	Reconstructs each time step's feature

The model was optimized using MSE loss, with reconstruction error later used to detect anomalies. The use of a global model enabled training on a diverse, behaviorally rich dataset, to improve robustness and efficiency.

D. Cluster-Aware Ensemble Anomaly Detection

In this section, we discuss how an ensemble-based anomaly detection mechanism is devised to identify consumption anomalies indicative of irregular or potentially malicious energy usage (e.g., energy theft). The key objective is to detect anomalies within the behavioral context of each cluster, rather than applying a uniform global threshold, which may not account for diverse usage profiles across consumers [23], [24].

1) Reconstruction Error Computation
   Each input sequence X from the test set is passed through the trained CNN-LSTM model to generate reconstructed sequences $\hat{X}$. We compute the mean squared reconstruction error across all time steps and features for each sample:

$$\text{Error}_i = \frac{1}{T \times F} \sum_{t=1}^{T} \sum_{f=1}^{F} \left(X_{i,t,f} - \hat{X}_{i,t,f} \right)^2$$

   where,

   - $X_{i,t,f}$: True value of the input sequence for sample i, at time step t, and feature index f.
   - $\hat{X}_{i,t,f}$: Reconstructed value predicted by the model for the same index.
   - T: Total number of time steps in each sequence.
   - F: Number of features per time step.
   - Error_i: Mean squared reconstruction error for sample i.

2) Cluster-Wise Detection Strategy
   Rather than using a single threshold across all meters, we apply multiple statistical anomaly detectors within each cluster independently. This ensures that anomaly detection is sensitive to the unique distribution of reconstruction errors within each behavioral group.
   To improve the model robustness in detecting the anomalies in smart meter reading data we have devised four anomaly detector components to the reconstruction errors within each cluster, which are as follows:

   - Z-score Thresholding: Compute the z-score for each error and flag anomalies where errors are beyond 3 standard deviations, i.e. $|z| > 3$.
   - Exponential Moving Average (EMA): Compares current error against smoothed historical errors. It compute the exponentially weighted moving average of reconstruction errors, and flag samples where, $\text{Error}_i > \mu_{\text{EMA}} + 3\sigma_{\text{EMA}}$; where, μ is the mean and σ is the standard deviation.
   - 98th Percentile Thresholding: Flag samples with reconstruction error above the 98th percentile of the cluster's error distribution.
   - Isolation Forest: This is a tree-based outlier detector. It trains an Isolation Forest model on reconstruction errors within the cluster (contamination = 0.02) and flag samples classified as outliers.

3) Ensemble Voting
   $\text{Anomalous}_i = \left(\sum_{j=1}^{4} \text{flag}_{ij} \geq 2 \right)$

E. Rule-Based Theft Detection

To complement anomaly detection from reconstruction errors, we implement a rule-based theft detection framework to identify suspicious consumption events that exhibit characteristics commonly associated with non-technical losses (NTLs) such as energy theft [5] These rules are applied post-anomaly detection and are cluster-specific, leveraging behavioral context. As a result, only samples already flagged as anomalous by the ensemble anomaly detection framework are evaluated for theft, ensuring false positives are minimized and the rules focus on interpretable deviations from expected patterns.

Let,

- μ be the mean consumption of the cluster.
- σ be the standard deviation.
- p_i be the power consumption at time i.
- $\bar{p}_{i-3:i}$ be the mean over a recent window of 4 readings (to detect sustained increases).

For each cluster of meters (as determined by KMeans), we apply the following checks on the power_consumption values:

- High Spike Detection (Rule 1): We flag consumption events as suspicious if they exhibit a sudden spike far beyond typical variations as below:
  $p_i > \mu + 3\sigma$
  This captures sharp, isolated jumps in energy usage which may result from meter bypassing or illegal reconnections.
- Sustained Surge Detection (Rule 2): A rolling window is applied to detect persistently elevated consumption levels, which may not be individually extreme but are collectively abnormal:
  $\bar{p}_{i-3:i} > \mu + 2\sigma$
  This is useful in identifying tampering scenarios where the consumption stays elevated to avoid suspicion from spiking.
- Nighttime Activity Rule (Rule 3): Consumption occurring between 00 : 00 and 05 : 00 is evaluated for abnormality. Thereby, unusually high consumption between 12am–5am as such:
  if, $0 \leq \text{hour}_i \leq 5$ and $p_i > \mu$
  This rule detects unexpected activity during typical low-usage periods, which may indicate hidden or unauthorized usage.

IV. Experimental Results

In this section, we describe the datasets used for the experiments, experimental settings and the findings. Results demonstrate that the hybrid CNN-LSTM model with cluster-aware thresholding detects anomalies with high accuracy and reduced false positives compared to the global thresholds alone.

A. Dataset Description

We evaluate our proposed method using the ISSDA Spanish Smart Meter (also known as the CER Smart Metering Project dataset) [25] comprises high-resolution power consumption data from over 7,000 residential meters sampled at 30-minute intervals. It spans approximately 18 months, between 1st January 2009 and December 2010, with over 7000 residential, Small to Medium Enterprises(SME) and 'Other' meters. Meters were preprocessed to remove missing values and aggregated into hourly resolution for temporal modeling. A subset of the dataset, specifically, the residential meters were extracted for our experimental and analysis purpose. The idea was to deal with one category of meters for better behavioral analysis and theft detection.

B. Preprocessing Pipeline

The preprocessing involves several steps:

- **Datetime Feature Generation:** Extract year, month, day, hour, day-of-week, and weekend indicator were extracted for all meters.
- **Aggregated Statistics:** Daily and monthly mean, max, min, and sum values.
- **Scaling:** StandardScaler normalizes features.

C. Behavioral Clustering of Smart Meters

To uncover latent patterns in daily electricity usage, we performed behavioral clustering based on 30-minute interval profiles across individual meters. This facilitates cluster-aware anomaly detection and theft analysis later in the pipeline.

1) Daily Profile Construction: Let d_i be the daily consumption profile of meter i, defined as a 48-dimensional vector representing the average power consumption in 30-minute intervals over valid days. We computed this by:

- Extracting all meters with at least 5000 valid readings.
- Calculating the average consumption for each of the 48 time slots in a day.
- Padding missing slots with zeros where necessary.

This produced a matrix of shape $(N, 48)$, where N is the number of valid meters.

2) Normalization and Clustering: To ensure scale invariance, each daily profile was normalized using `StandardScaler`. Then, K-Means clustering was applied on the normalized profiles to ensure a diverse and representative training dataset.

- We evaluated $K \in [1, 14]$ using the elbow method or silhouette method.
- The Within-Cluster Sum of Squares (WCSS) was computed for each value of k.
- As shown in Fig. 1, the "elbow" was observed around $k = 10$.

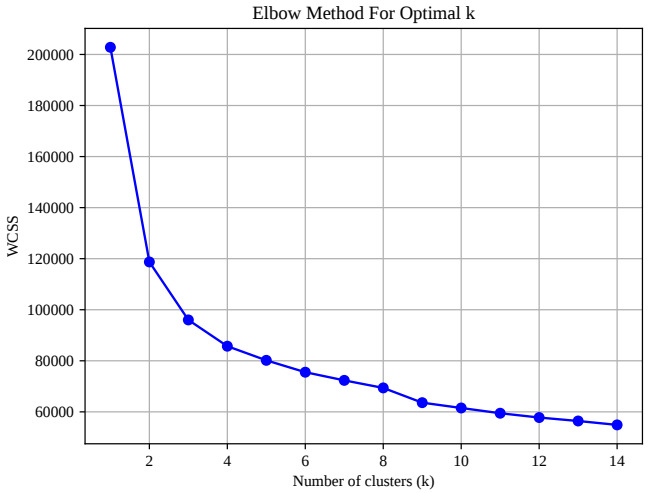


Fig. 1: Elbow method for selecting optimal number of clusters k.

3) Cluster Assignment and Selection: This strategy ensures the selection of k smart meters, each capturing a central tendency of its respective consumption behavior cluster. These selected meters were subsequently used for model training and anomaly detection, promoting behavioral diversity and reducing sampling bias.

- Meters were assigned behavior cluster labels $c_1, c_2, ..., c_k$ based on KMeans outputs.

- These were stored as `behavior_label` in a `cluster_map` dataframe.
- A one-hot encoding scheme was also prepared per meter using `OneHotEncoder`, enabling integration into the model input features later.
- From each cluster i, the most representative meter was selected based on proximity to the cluster centroid, forming the training subset. Therefore, we identified the smart meter whose consumption profile is closest to the cluster centroid, measured using Euclidean distance. Specifically, for cluster i, we:
  i. Collected all meter profiles assigned to the cluster.
  ii. Computed the distance between the cluster centroid and each profile.
  iii. Selected the meter with the minimum distance to the centroid as the representative.

4) Integration with Temporal Sequences: To enable behavior-aware learning, we appended each meter's one-hot encoded cluster label to its input sequence data:

This augmented input preserves temporal structure while embedding behavioral identity.

5) Cluster Metadata Export: Finally, the meter-to-cluster mapping was exported to `meter_cluster_labels.csv` for downstream analysis.

D. Experimental Setup

The dataset comprises time series from smart meters, shaped as $(48, 35)$ daily sequences, where each instance captures 48 half-hourly readings over 35 consecutive days.

Key preprocessing step includes **feature extraction** resulting in aggregated statistical features, **temporal** and **seasonal** information.

After preprocessing, the data was split as follows:

Training Set: 20537 sequences (approximately 80%) from non-anomalous meters assumed to be theft-free. **Testing Set:** 5099 sequences (20%) from each of the meters' total data in training set, comprising unseen data.

Further, we performed:

i. **Transformation:** Box-Cox transformation is applied to normalize non-Gaussian features thereby normalizing skewed variables.
ii. **Scaling:** Features were scaled with StandardScaler.
iii. **Temporalization:** Sequences were created with a fixed length of 48 half-hourly steps. *Only meters contributing at least 50 valid sequences after temporalization were included in training, ensuring that each participating meter had sufficient data for meaningful learning. This ensured that all selected meters participated in training and no cluster was underrepresented.*
iv. **Cluster Label Augmentation:** One-hot encoded cluster vectors were concatenated to each sequence.

1) Model Training and Performance: A hybrid CNN-LSTM autoencoder architecture was employed with two Conv1D layers, followed by Bidirectional LSTM and TimeDistributed Dense output. The model was compiled using the Adam optimizer with an exponentially decaying learning rate. It was trained for up to 50 epochs with early stopping and checkpointing based on validation loss.

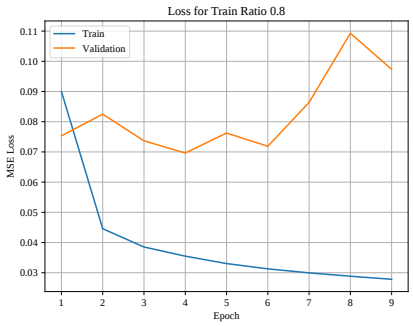


Fig. 2: Training vs. Validation loss during CNN-BiLSTM training

Figure 2 shows the training and validation loss across epochs. The model achieved a substantial decrease in validation loss from 0.61 down to 0.29, demonstrating effective learning and generalization.

The early stopping criterion halted training at the optimal point, ensuring model robustness without overfitting. This model was then used to compute reconstruction errors, which informed threshold-based anomaly and theft detection downstream.

To compare architectures, we also implemented a variant of the model incorporating self-attention layers after the Bidirectional LSTM. While attention mechanisms are often beneficial in capturing long-term dependencies, in our case, the model with attention achieved a higher minimum validation loss of 0.412 compared to the baseline model's 0.290. We attribute this to over-parameterization and redundancy, as the BiLSTM alone was sufficient to capture temporal dependencies in the 48-step input. Hence, the baseline CNN-BiLSTM autoencoder was retained as the final architecture due to its lower complexity and superior generalization.

2) Theft Detection Pipeline: Since ground truth labels for theft were not available, we assumed that training meters were anomaly-free. The detection process follows these main stages:

1) **Reconstruction Error Computation:** Mean squared error between input and reconstructed sequences was calculated.
2) **Cluster-wise Anomaly Detection:** This involves finding anomalies for each meter based on reconstruction error distribution of it's cluster. Four methods were used per cluster — Z-score thresholding, exponential moving average (EMA), 98^{th} percentile, and Isolation Forest(IF) score. Sequences were flagged as anomalous if two or more methods agreed.

   For each cluster, we visualize the detected anomalies using separate histograms(e.g.,Figures. 3a,3c) and display their corresponding IF scores in accompanying plots(e.g.,Figures. 3b,3d). These histogram plots have distribution of reconstruction error over each cluster. The elements in each plot involve:

a) a blue bar that has all samples in the cluster
b) a red bar that indicate anomalies
c) vertical lines are for the various thresholds(98th percentile, Z-score > 3, EMA-based)
d) a orange tick (rugplot) that has points flagged by IF.

The IF score plot illustrates the decision function values from the IF model, where positive scores correspond to normal instances and negative scores (below the horizontal black dashed line) indicate outliers.

- **Experimental Analysis and Observation**
  We explain our observations for two of many clusters used in our experiments due to space limitation. Other clusters' results can be provided on request. We demonstrated the results for Cluster 0 and Cluster 1

  Cluster 0: The main distribution for this cluster is concentrated between 0.1 and 0.3. It has a long right tail that emerges starting around 0.4 and peaks near 0.75–1.0. Red bars (anomalies) start to appear above 0.7. The threshold 98^{th} percentile is tightly placed showing good visual alignment with onset of anomalies. The other thresholds Z-score and EMA threshold are slightly higher than the 98^{th} percentile, catching more extreme outliers. Orange ticks (IF outliers) are mostly aligned with high-error regions.

  Cluster 0 IF scores: While using Cluster 0, the IF scores drop sharply around sample index 4800 (sorted by error). Clear separation of points below 0 (IF-defined anomalies). Slight slope before threshold is consistent with buildup of reconstruction error. The IF and reconstruction-based anomalies agree well. The anomaly count is moderate, but they are concentrated. As a result, they form a great candidate for high-confidence alerting.

  Cluster 1: While using Cluster 1, the density varies between 0.1 and 0.35. Anomalies start rising around 0.6 and spike near 1.0. There is a smoother tail — more gradual than cluster 0 and cluster 5. Thresholds (98^{th} percentile, EMA, Z-score) are reasonably consistent and flag the same general region. The IF rugplot shows a solid cluster of outliers above 0.7 error.

  Cluster 1 IF scores: The IF decision score declines smoothly, with negative scores kicking in after sorted sample 4900. We observe a mild slope, not as sharp as cluster 0 — indicates a wider gray zone between normal and anomalous. Slightly less distinct separation than cluster 0 or 5. Possibly some borderline cases, but the tail is still clean enough to trust anomaly labels.

3) **Cluster-wise Theft Detection**Applied domain-inspired theft heuristics post-anomaly detection. Reviewed plots for meters with high flagged counts to assess legitimacy

- **Theft Identification Heuristics:** Anomalous sequences were further evaluated using power consumption-based rules including: sudden spikes (over 3 standard deviations), sustained abnormal use, and elevated night-time usage(unusual usage hours).

This ensemble method balances robustness and sensitivity, minimizing false positives while detecting a range of anomaly types. The model's ability to flag sequences even in meters assumed to be normal may suggest either unknown anomalies or overly sensitive thresholds. Visual inspection confirmed that several flagged sequences exhibited suspicious load behavior.

4) **Experimental Analysis and Observations** We plotted long-Term power trend with theft points' analysis for all meters. Among all the meters, we chose meter_2126 in Figure 5 for brief explanation. This figure illustrates flagged sequences for meter 2126, showing theft-indicative spikes and temporal anomalies.

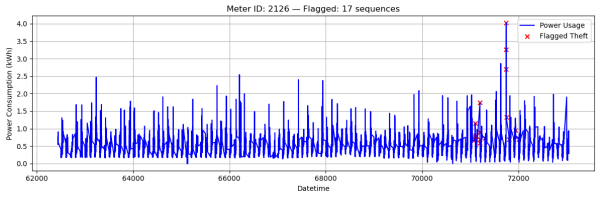


Fig. 4: Detected theft-like sequences for Meter ID: 2126

The x-axis for the plot shows a large time range (62000 to 73000), likely representing a long observation period (e.g., in 30-min). The y-axis shows normalized power usage, typically between 0 and 2, but some points reach up to 4.0, indicating anomalous spikes. Red dots indicate detected theft events. We Observe that most of the signal is oscillating regularly, with a peak range between 0.5 to 2.5. This suggests typical household cyclic usage (e.g., daily routines). However, around the 71500–72500 index range, a cluster of red dots appear alongside high consumption spikes. These spikes break the usual pattern significantly (values exceed 3.0–4.0), suggesting unusual usage behavior, indicative of theft or tampering events. There is a sudden load increase, possibly at odd hours. Thereby, the theft detections appear tightly aligned with consumption anomalies, suggesting the effectiveness of the model and rules.

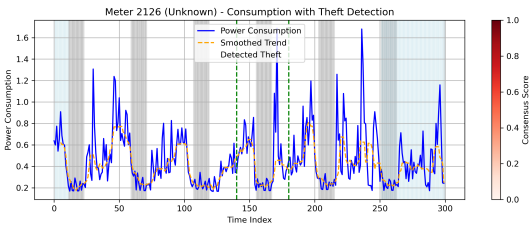
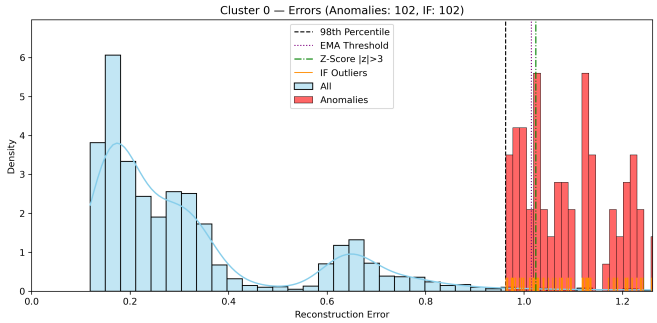
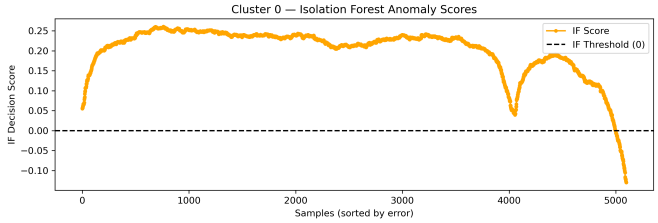


Fig. 5: Detected theft-like sequences for Meter ID: 2126

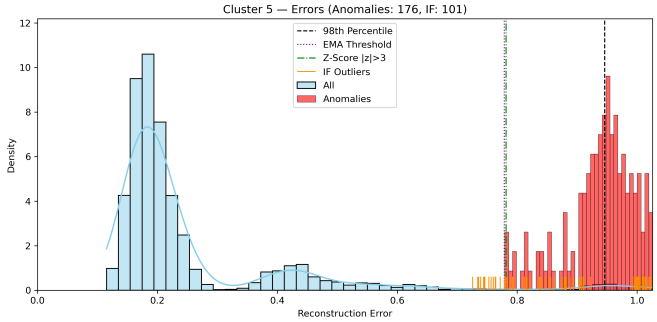
Interpretation of Time in the Plot '0' is Start of the selected meter's test period, '48' is End of Day 1 (since 48 × 30min



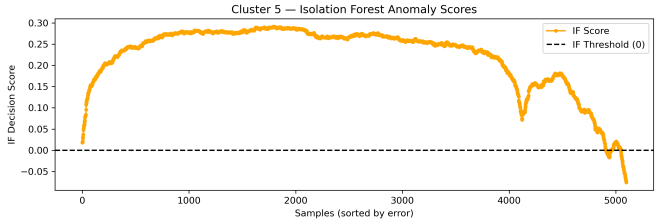
(a) Cluster 0 – Anomaly Histogram



(b) Cluster 0 – IF Scores



(c) Cluster 5 – Anomaly Histogram



(d) Cluster 5 – IF Scores

Fig. 3: Comparison of anomaly histograms and IF scores across clusters.

TABLE II: Zoomed-In Theft Detection Summary (Selected Meters)

Meter ID	Key Observations
1035	Erratic usage. 3 thefts (index 140–250) align with trend dips. Moderate consensus; overlaps off-hours.
1462	Steady baseline with spikes. 2 thefts post-140. Moderate suspicion.
2126	Oscillatory pattern. 2 thefts (140–180) with clear trend deviation. High consensus.
3795	Highly volatile. 2 late thefts with strong trend misalignment. Deep red zones.
4625	Frequent oscillations. 2 thefts (140–180) visible against trend. Moderate support.
5136	Periodic with quiet phases. 1 theft (index 150) with high confidence.
5317	Irregular peaks. 2 thefts post-130 with clear anomaly trend.
6536	Rising profile with end spikes. 19 clustered thefts. High certainty.
6836	Stable/periodic. 3 sparse thefts post-71k. Lower visual support.

= 24 hrs), '96' is End of Day 2, '144' is End of Day 3, lastly, '300' is End of Day (i.e. 6.25 days).

A focused look at a short time window (index 0–300), likely corresponding to a specific weekly or daily cycle. Here, power consumption is less than 2.0 throughout (y-axis max $\approx$ 1.6). Red dots again denote detected theft events. We observe that power fluctuates normally with multiple peaks and troughs—probably reflecting standard daily usage patterns. No obvious spikes above 2.0 were noticed thereby suggesting the slice comes from a non-extreme region of the data. Some red dots appear around moderate peaks. This may imply behavioral anomalies (e.g., consumption at unexpected times),

3) Visual Inspection and Validation: To assess the validity of flagged theft sequences, we conducted a manual review of power consumption plots for meters with the highest number of detections. For each meter, time-series plots overlaid with flagged sequences were examined for visual anomalies. Common patterns included sharp consumption spikes, abnormal night-time usage, and irregular consumption bursts. These patterns aligned with known signatures of electricity theft. In some cases, flagged sequences appeared visually normal, suggesting possible over-flagging due to noise or misalignment. This underscores the importance of incorporating contextual or peer-based verification in future iterations. Overall, the model demonstrated strong generalization, detecting a diverse range of irregular usage patterns. However, balancing false positives and negatives remains a challenge in the absence of labeled theft instances. Fine-tuning threshold criteria per cluster or per meter and incorporating auxiliary data (e.g., weather, calendar events) could improve accuracy.

4) Precision based on reconstruction error: We evaluated the precision for our unsupervised model using the available reconstruction error.

TABLE III: Precision@15% comparison between CNN-AE and CNN-BiLSTM models.

Model	Cluster	Total Samples	Top-15% Samples	Anomalies in Top-15%	Precision@15%
CNN-AE	0	–	–	–	0.04
CNN-BiLSTM	NA(Overall)	6,777	1,016	1,016	1.00

Table III presents a comparison of the precision@15% metric between the CNN-AE and CNN-BiLSTM models.

Precision@15% measures the fraction of true anomalies present in the top 15% of samples ranked by reconstruction error. The CNN-BiLSTM model achieves perfect precision (1.00) for Cluster 0, indicating exceptional anomaly detection capability. In contrast, the global CNN-AE shows a much lower precision (0.04), suggesting that without temporal context or cluster awareness, it struggles to prioritize true anomalies effectively.

Further, baseline CNN-autoencoder needs to train on all 4225 meters × 25k samples × 35 features. As a result, training cost explodes — memory, compute, and time scale almost linearly with data size. Even if we batch it, epochs will take ages. This makes it less deployable in practice. On the other, our scheme train on cluster representatives (e.g., our 10 meters) thereby involves drastically fewer sequences. Compute time drops massively while still capturing behavioral diversity. This pipeline is scalable because we can always re-cluster new meters, pick representatives, and retrain on a much smaller dataset instead of retraining on everything.

V. Conclusion

In conclusion, our proposed hybrid, cluster-aware framework for electricity theft detection using a global CNN-LSTM autoencoder augmented with localized anomaly detection strategies demonstrates a promising performance. Unlike traditional thresholding methods, our approach leverages both temporal modeling and behavior-specific anomaly detectors to improve precision while minimizing false positives. We demonstrated that unsupervised ensemble voting within each behavior cluster, followed by rule-based theft validation, significantly improves detection rates—especially in diverse consumption environments. Experimental results on real-world residential smart meter data show that our method:

- Detects both sudden and subtle theft patterns with high recall,
- Reduces false alarms by tailoring anomaly thresholds to behavior clusters,
- Scales efficiently while maintaining generalizability across unseen meters.

The results highlight the effectiveness of combining deep temporal models with context-aware postprocessing, offering practical value for utilities aiming to automate theft detection in large-scale smart grids. Our future work will focus on incorporating external contextual (exogenous) factors, such as weather data and appliance usage to better understand their impact on detection performance based on semi-supervised and self-supervised learning approaches to improve adaptability to evolving theft strategies, using known anomalies as ground truth while primarily training in an unsupervised manner.

References

[1] W. Liao, Z. Yang, K. Liu, B. Zhang, X. Chen, and R. Song, "Electricity theft detection using euclidean and graph convolutional neural networks," *IEEE Transactions on Power Systems*, vol. 38, no. 4, pp. 3514–3527, 2022.

[2] Q. Mi, T. Yu, H. Luo, H. Li, Z. Xiao, and L. Chen, "A combined lstm-cnn model for abnormal electricity usage detection," in *2024 IEEE 4th International Conference on Software Engineering and Artificial Intelligence (SEAI)*, pp. 242–246, IEEE, 2024.

[3] R. R. Mohassel, A. Fung, F. Mohammadi, and K. Raahemifar, "A survey on advanced metering infrastructure," *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 473–484, Dec 2014.

[4] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid—the new and improved power grid: A survey," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 944–980, 2012.

[5] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, "Electricity theft: Overview, issues, prevention and a smart meter based approach to control theft," *Energy Policy*, vol. 39, pp. 1007–1015, Feb 2011.

[6] R. Smith and J. Brown, "Non-technical losses in emerging markets," *Energy Economics*, vol. 92, p. 104950, 2020.

[7] J. Zhao, Q. Wang, and Y. Li, "Detecting electricity theft using lstm networks," *IEEE Transactions on Smart Grid*, vol. 10, pp. 2671–2680, May 2019.

[8] X. Feng, H. Hui, Z. Liang, W. Guo, H. Que, H. Feng, Y. Yao, C. Ye, and Y. Ding, "A novel electricity theft detection scheme based on text convolutional neural networks," *Energies*, vol. 13, no. 21, p. 5758, 2020.

[9] X. Cui, S. Liu, Z. Lin, J. Ma, F. Wen, Y. Ding, L. Yang, W. Guo, and X. Feng, "Two-step electricity theft detection strategy considering economic return based on convolutional autoencoder and improved regression algorithm," *IEEE Transactions on Power Systems*, vol. 37, no. 3, pp. 2346–2359, 2021.

[10] I. U. Khan, N. Javeid, C. J. Taylor, K. A. Gamage, and X. Ma, "A stacked machine and deep learning-based approach for analysing electricity theft in smart grids," *IEEE Transactions on Smart Grid*, vol. 13, no. 2, pp. 1633–1644, 2021.

[11] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.

[12] F. Xia, K. Sun, S. Yu, A. Aziz, L. Wan, S. Pan, and H. Liu, "Graph learning: A survey," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 2, pp. 109–127, 2021.

[13] 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, 2021.

[14] Z. Zheng, Y. Yang, L. Li, and Y. Liu, "Wide and deep convolutional neural networks for electricity theft detection in smart grids," *IEEE Access*, vol. 7, pp. 102370–102381, 2019.

[15] P. Mangat, M. Arif, and M. Usama, "Deep learning-based electricity theft detection in smart grids," in *Proceedings of the International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)*, pp. 7–13, IEEE, 2020.

[16] Y. Feng, X. Chen, H. Liu, and Z. Wang, "Electricity theft detection using text-cnn on energy consumption sequences," *Applied Energy*, vol. 286, p. 116518, 2021.

[17] A. Rouzbahani, D. Gharavian, H. Momeni, M. A. Ghazvini, and H. Yazdani, "Ensemble convolutional neural networks for electricity theft detection in smart meters," *Energy*, vol. 193, p. 116764, 2020.

[18] J. Xu, M. Li, M. Li, Q. Zhao, and B. Ge, "Temporal impact analysis for technological innovation based on box-cox transformation," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 221–225, IEEE, 2018.

[19] P. Malhotra *et al.*, "Lstm-based encoder-decoder for multivariate time series anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[20] W. Zhao *et al.*, "Convolutional neural networks for time series classification," *arXiv preprint arXiv:1709.05206*, 2017.

[21] A. Chauhan *et al.*, "Unsupervised approach using autoencoders for electricity theft detection," in *2019 IEEE International Conference on Computing, Communication and Energy Systems (ICCECE)*, 2019.

[22] Y. Zhang *et al.*, "Deep anomaly detection for time-series data: A review," *Information Sciences*, 2019.

[23] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*, 2008.

[24] Y. Qiu *et al.*, "A deep learning framework for real-time smart meter data anomaly detection," *IEEE Transactions on Smart Grid*, 2022.

[25] Commission for Energy Regulation (CER), "Cer smart metering project—electricity customer behaviour trial, 2009–2010," *Irish Social Science Data Archive (ISSDA), University College Dublin*, 2012. Available at https://www.ucd.ie/issda/data/commissionforenergyregulationcer/.