JJEE

Volume 11 | Number 3 | September 2025 | Pages 515-528

Jordan Journal of Electrical Engineering

ISSN (print): 2409-9600, ISSN (online): 2409-9619

Homepage: jjee.ttu.edu.jo

Anomaly Detection Framework for Smart Grids Using Deep Machine Learning and Extreme Gradient Boosting Algorithms

Zeyad A. Al-Odat[1], Eman Al-Qtiemat[2], Alaa A-Quteimat[3],

Abdullah Eial Awwad[4*]

[1,] Department of Computer and Communication Engineering, Tafila Technical University, Tafila, Jordan
[2] Department of Computer Science, Tafila Technical University, Tafila, Jordan
[3,4] Department of Power and Mechatronics Engineering, Tafila Technical University, Tafila, Jordan
E-mail: abdullah.awad@ttu.edu.jo

Received: Mar 10, 2025 *Revised: May 17, 2025* *Accepted: Jun 25, 2025* *Available online: Nov 27, 2025*

***Abstract*—** The increasing demands for supervisory power management systems transform traditional power management systems into smart grids. To achieve this, management, information, and communication technologies (ICT) are integrated into power grids via the use of smart grid technology. This integration empowers consumers and providers of electrical utilities, enhances the efficiency and reliability of the power system, guarantees management continually, and controls client demands. However, smart grids are distributed in vast and complex networks, including millions of interconnected computers, devices, and other components. This distribution exposes the extensive power networks to several security flaws and breaches. This work presents a deep learning algorithm-based anomaly detection framework for smart grids. Extreme gradient boosting (XGBoost) is coupled with long-short term memory (LSTM). Based on high accuracy values, the proposed design is shown to be able to detect and identify abnormal behaviors. Furthermore, an extra evaluation for the suggested design using other performance criteria, including mean square error, F1-score, precision, recall, and root mean square error.

***Keywords*—** Smart grids; Security; Anomaly detection; Machine learning; LSTM; XGBoost; DNN.

1. INTRODUCTION

Information and communications technology (ICT) is applied in data flow control and management towards the grid [1]. The ICT system helps energy companies to precisely control and handle power consumption. Smart grids give consumers' digital communication with energy providers the significance [2].

The build and control of a safe automated supervising control and data acquisition system (SCADA) depends on the security of smart grids. Any kind of risk that targets the smart grid system will directly impact consumers and companies, so rendering them vulnerable and compromised [3]. Three main components—confidence, integrity, and availability—which together form the CIA triad—must be included in the design if one is to achieve secure systems, especially smart grids [4].

Smart grids suffer from different vulnerabilities [5], which include:

1) Insufficient authentication,
2) Inadequate data encryption,
3) Lack of continuous data integrity.

These vulnerabilities expose the smart grid infrastructure to various threats and assaults. Therefore, escalated security measures and prevention and detection mechanisms are needed

[*] Corresponding author

DOI: https://doi.org/10.5455/jjee.204-1741611858

to enhance the smart grid security measures and comply with the general CIA triad requirements [6]. Moreover, continuous human awareness training and updates on safety measures are essential for preventing electrical data assaults. A common daily error may serve as a straightforward catalyst for a data breach. Common human errors include clicking on incorrect spam links, disclosing confidential information to unauthorized individuals, and generally neglecting security policies [7].

The primary components of a smart grid, as reported by the NIST, are electrical household appliances, renewable energy resources, smart meters, electric utility operation centers, and service providers [6, 8]. Electrical household appliances, both smart and traditional, are presumed to interact with the smart meter IoT network, enabling effective control of power usage across all household devices. Renewable energy resources include solar and wind energy, which provide locally generated power for household appliances. Smart meters regularly record power use, communicate data to the utility computer, manage customer power supply connections, and provide warnings in the event of anomalies [9]. Certain smart meters include relays that may interface directly with smart home equipment for control purposes; for instance, to deactivate the air conditioner during peak hours. Moreover, the smart meter may be used in demand-side control. The Electric Utility Center engages with smart meters to manage power use. It transmits consumption-related directives to smart meters and gathers sub-hourly power use records together with emergency/error warnings [10].

Machine learning algorithms were extensively used in the field of anomaly detection. Machine learning algorithms aim to analyze streaming data to create models that can be utilized with parameters and across various applications [11, 12]. In practical scenarios, analyzing a substantial quantity of sensor values concurrently renders detailed knowledge of individual parameters largely secondary [13]. Anomaly detection algorithms for manufacturers should require minimal parameter settings to facilitate ease of application and suitability for streaming analysis. Therefore, these algorithms must incorporate mathematical operations characterized by low computational complexity to reduce latency in extended streaming analyses [14].

Deep learning is a subset of machine learning that facilitates the acquisition of effective data representations via several levels of abstraction [15]. This allows deep learning to surpass conventional machine learning methods as the volume and diversity of data expand [16]. Recently, deep learning-powered anomaly detection techniques have gained significant use across several fields. This work presents a hybrid machine learning framework for the detection of anomalous actions in smart grid infrastructure. The proposed design employs the Long-Short Term Memory (LSTM) deep learning algorithm for feature extraction and the Extreme Gradient Boosting (XGBoost) algorithm for classification.

The contributions of the proposed model can be outlined as follows:

1. An optimized LSTM-XGBoost framework is proposed for the detection of electricity anomalies. This framework integrates XGBoost classification power with LSTM efficiency in dealing with time-series datasets.
2. The dataset is preprocessed for better interpretation, then fed to the LSTM for feature extraction.
3. High accuracy results are achieved through the implementation of the XGBoost classifier.

The rest of the paper's sections are organized as follows. Section 2 presents the literature works in the same area of study. Section 3 provides detailed explanations about the proposed design. Section 4 presents the achieved results and discusses the outcomes. Section 5 concludes the paper.

2. LITERATURE REVIEW

Due to the increasing complexity of smart grids, efficient, secure, and dependable infrastructure must be found [17]. The statistical method in detecting anomalies lacks the proper interpretation of the time series dataset. Therefore, machine learning algorithms are employed to handle the dynamic and high-dimensional nature of the time-series dataset. These algorithms are used to analyze and detect improper actions or misuse cases, particularly anomalous assaults. Recently, deep learning algorithms, such as LSTM and XGBoost, have innovated in modeling and classifying the temporal nature of time-series applications, e.g., smart grid datasets [18]. In the subsequent text, we present current methodologies in the area of anomaly detection for smart grids.

Among the non-technical losses faced by an electrical supply, anomalous actions represent one of the most significant issues in smart grid environments. Improper use of power leads to degradation in the quality of power supply, leading to escalated power production. Therefore, forcing the legitimate customers to accommodate high costs. A convolutional neural network (CNN) with a long short-term memory model was suggested by Hassan *et al*. In their work, they employed CNN to extract the features of misbehavior actions and classify them using LSTM. Their work employed under-sampling data balancing algorithms and provided accuracy before and after data balancing. The reported results presented around 90% accuracy before and after optimization, achieving no big difference [19].

The unbalanced dataset in the domain of misuse renders several artificial intelligence methods prone to underfitting. To address this problem and detect several sorts of stealing attacks, the authors suggested a methodology using local outlier factor (LOF) and clustering (Peng et al., 2021). In their work, k-means clustering is used to assess the load data. Then, the LOF is used to find the degrees of options for outliers that don't seem right. After that, a similar structure for real execution is set up. The method's effectiveness is finally proven by numerical tests on real datasets [20].

The work in [21] presented a model that employs deep neural network algorithms for the detection of power anomalies. A three-stage framework was developed, including selection, extraction, and categorization of characteristics. During the selection process, the average hybrid feature significance identifies the most significant characteristics and their priority level. The feature extraction methodology utilizes the ZFNET algorithm, which helps in eliminating extraneous features. The suggested framework used a hybrid approach that combines the two deep learning algorithms convolutional neural network (CNN) and long-short term memory (LSTM). The hyperparameters of the DNN models were configured using black widow and blue monkey optimization models. Optimizing the classifier's configuration hyperparameters enhances data training efficacy. Following comprehensive simulations, their work suggested methodologies CNN-LSTM-BlueM and CNN-LSTM-BlackW achieved overall system accuracies of 91% and 93%. The efficacy of the proposed framework achieved elevated accuracy and a minimal mistake rate.

Other work suggested system models that employed normalization and interpolation techniques to preprocess the electrical data. The chosen characteristics are thereafter sent to the RUSBoost frame for analysis, detection, and classification. The simulation results proved that the proposed design was able to address the issue of an imbalanced dataset. Moreover, the framework showed enhanced behavior in dealing with time-series datasets. The authors claimed that their proposal surpassed other baseline machine learning algorithms, such as SVM and LR, and deep learning algorithms, such as CNN. Their framework was tested using performance measures and performance metrics: recall, F1-score, and precision, in addition to the receiver operating characteristic curve (ROC) [22].

The work [23] proposed an anomaly detection approach utilizing end-to-end internet performance measurements. The proposed work employs cluster-based local outlier factor (CBLOF). The authors claim that the input dataset suffers from missing values due to congestion and anomalous spikes. Their work was targeted to detect anomalous behaviors of outliers. The k-nearest neighbor (kNN) is used to fix the missing values from the dataset. Then the design divides the input data into clusters for better representation. The results showed that some clusters have a high anomaly rate compared to other clusters.

The work in [24] examines the issue of misclassification resulting from cross pairings. A cross pair is an intersection of two instances from opposing classes. The Tomek linkages approach is used to identify these cross pairings. The samples from the majority class linked to cross pairings are eliminated to distinguish the two opposing classes using an affine boundary call. In the absence of anomaly, six anomalous instances are used to generate synthetic anomalous data that simulates a real-world situation. These six assaults pertain to innocuous class data, whereby benign samples are altered, and harmful samples are generated. Additionally, K-means SMOTE is used to address the class imbalance problem by generating balanced data. Furthermore, the technical approach involves training the model using time-series data from both groups. Training a model using imbalanced data often leads to misclassification of samples owing to bias toward the dominant class, resulting in a high false positive rate.

The work in [25] applied machine learning algorithms Decision Tree (DT), Random Forest (RF), and Multi-Layer Perceptron (MPL) on a time-series sensor dataset. The proposed design compares the results of utilizing these algorithms and compares them regarding precision and true positive rates. The analyses demonstrated effective management of machine learning and its components in industrial manufacturing settings, which can be significantly enhanced by anomaly detection.

The concept of temporal convolutional neural network gained the attention of authors in [26]. The proposed work focuses on the spatial nature of the dataset instead of the time-series property. An anomaly detection approach based on spatio-temporal learning was proposed. The proposed design comprises three components: 1) graph convolutional neural networks (GCN), 2) multi-scale CNN network, and 3) combined spatial and temporal features. The reported results of the F1-score and AUC of the user electricity consumption data collected by the State Grid Corporation's smart meter are 0.935 and 0.977, respectively. The model demonstrates excellent stability when confronted with extreme data imbalances. It is generalizable through experiments and can be applied to other datasets. On the other hand, the work [27] suggested method utilizes SVM machine learning to detect fraudulent behavior.

The assessment of the suggested method using an actual power usage dataset demonstrates a high detection rate and a low false positive rate in comparison to similar studies.

The work presented by [28] proposed a trace-based graph deep learning algorithm to detect smart grid anomalous customers. The suggested model uses an unsupervised encoder–decoder machine learning model. In their work, the authors merged traces into a uniform graph and delivered quality values. The LSTM framework extracts the temporal features, while the graph neural network (GNN) is employed to extract the spatial features. The final anomaly score is calculated by adding two hyperparameters' configurations with two-part losses. The reported results showed that the work outperformed several traditional anomaly detection models with a 94.60% F1 score and a 98.90% AUC.

Another work influenced by the deep learning algorithms, as the work presented in [13]. This work presented an anomaly detection framework based on deep learning for IoT. The proposed design captures the robust features of the IoT infrastructure through dynamic learning. The captured features were used to detect anomaly in the IoT data by training a denoising autoencoder deep learning algorithm. The reported experimental results showed an effectiveness in the detection of unusual action on the IoT with around 95% accuracy.

The majority of current research complies with conventional machine learning methodologies, which are time-intensive for training and prone to high error rates. There is a need to use deep learning methodologies such as LSTM. Our suggested methodology consists of three stages: data pre-processing, feature extraction using LSTM, and classification using the XGBoost model.

3. PROPOSED METHODOLOGY

A) Problem Description

Electricity is generated and distributed by power networks. The power grids are dispersed networks situated near energy sources. The grids use the energy to generate power. The power grid network comprises smart grids that facilitate the transmission of generated energy to customers. Smart grids are equipped with intelligent sensors or meters that collect and store data on power use and statistics. Smart grids integrate the functionalities of big data and edge computing.

Smart meters, which facilitate the analysis of substantial data generated by intelligent sensors, are thus susceptible to malicious assaults. These assaults often include hacking the electric meter, misconfiguring the meter, or circumventing the electric meter to provide fraudulent readings. These erroneous readings adversely impact the efficiency, quality, and transmission of power. Consequently, the security of smart grids is of supreme importance among other characteristics of smart grid components. Therefore, this work proposes an LSTM with an XGBoost model to address these issues.

B) System Architecture

The proposed framework consists of three primary components: data processing, feature extraction and classification, and evaluation, as depicted in Fig. 1. In the following, we summarize these components, and more details will be included in the subsequent subsection.

1) Data Preprocessing: This stage enhances data quality and dependability by rectifying problems such as incompleteness, noise, and inconsistencies, hence facilitating informed decision-making. In the proposed design, we used the interpolation technique. This technique operates on the idea of arithmetic mean, wherein it substitutes the missing

value with the average of its preceding and succeeding values. Then the normalized data is divided into training and testing parts.

2) Feature extraction utilizing the LSTM model: As a variation of the recurrent neural network (RNN), the LSTM network includes gate units into the state dynamics in order to address the gradient vanishing issue that is present in RNN. One of the reasons why LSTM is suited for dynamic predictive modeling is because it has both long-term and short-term dynamic memory on its own. In this stage, the LSTM is used to extract the temporal features of the hidden input data window.

3) Classification and evaluation employing the XGBoost model: The adoption of XGBoost is due to its capability to manage huge datasets quickly and efficiently, facilitated by rapid training using parallel processing. XGBoost facilitates the comprehension of feature relevance, helping in the selection of features based on their significance. The features extracted from the previous stage are used to train XGBoost and classify the sequences of the input data (normal/anomalous).

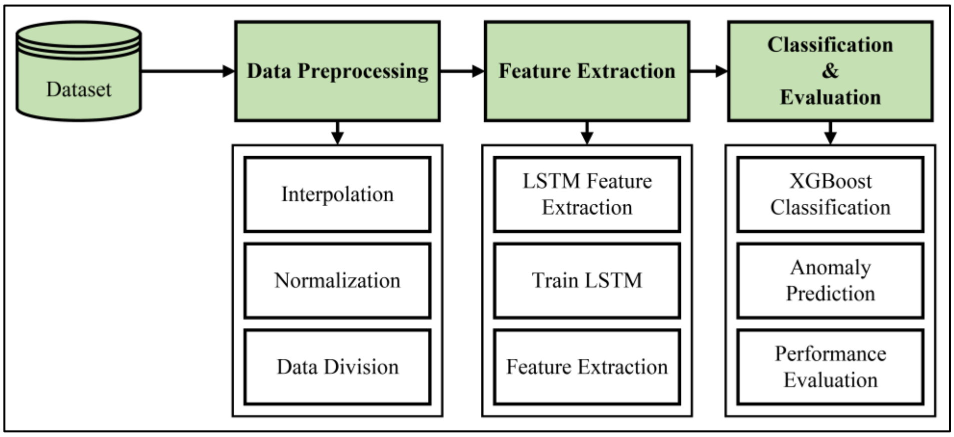

Fig. 1. Proposed architecture.

C) XGBoost Methodology

Extreme Gradient Boosting (XGBoost) is an enhanced technique derived from Gradient Boosted Decision Trees (GBDT), introduced by [29]. The distinguishing characteristics of XGBoost that set it apart from other gradient boosting algorithms include the insightful consequence of trees, proportional reduction of leaf nodes, and an additional randomization parameter. This research uses the XGBoost algorithm to extract features and assess their relative relevance. XGBoost's internal architecture is depicted in Fig. 2.

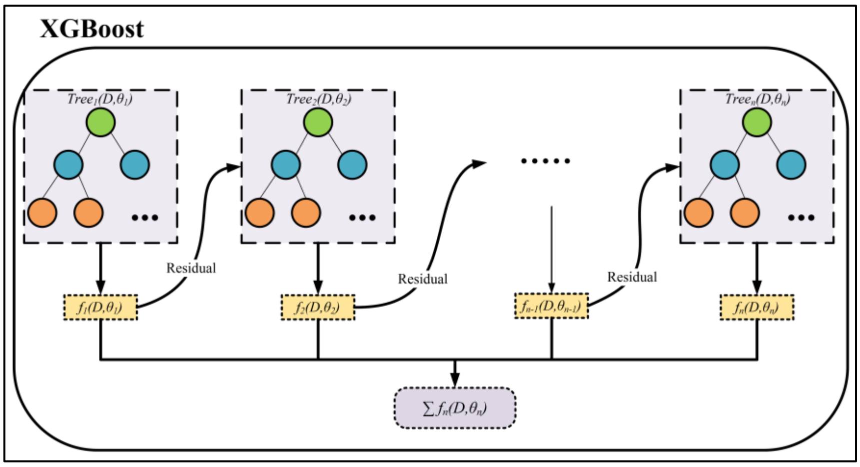

Fig. 2. The internal structure of the XGBoost algorithm.

Every iteration of XGBoost is characterized by a series of trees. The quantity of trees is established during the hyperparameter setting phase for the classifier. With each iteration, an additional tree is included to reduce the residual error from the preceding tree. The output of XGBoost is the aggregate of projections from all trees, which is calculated using Eq. (1).

$$\hat{y}_i = \sum_{t=1}^{T} f_t(x_i) \tag{1}$$

where f_t: prediction i , x_i: input stream data, and T: trees number.

D) LSTM Algorithm

As a variation of the recurrent neural network (RNN), the LSTM network includes gate units into the state dynamics in order to address the gradient vanishing issue that is present in RNN. One of the reasons why LSTM is suited for dynamic predictive modeling, power grids in our case, is because it has both long-term and short-term dynamic memory on its own. LSTM internal architecture is depicted in Fig. 3.

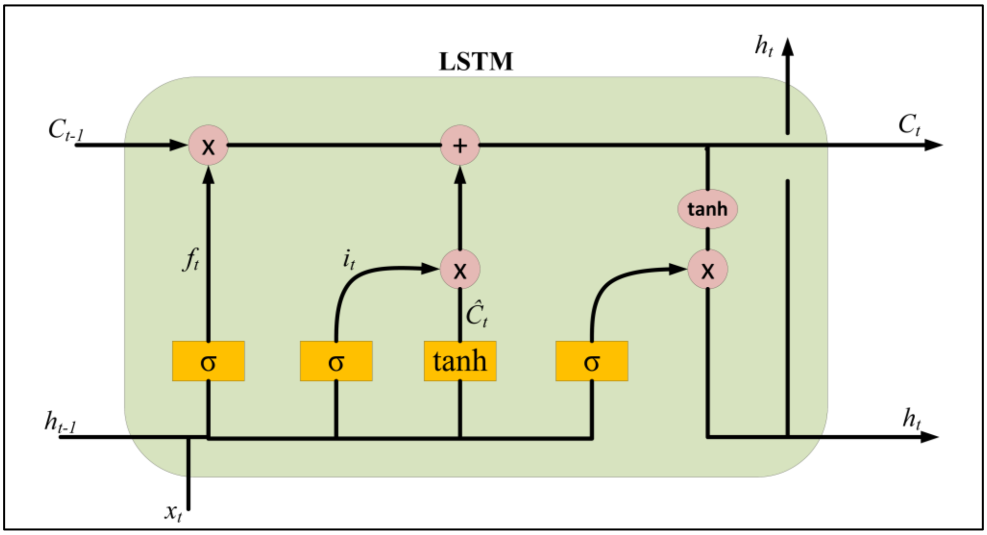

Fig. 3. LSTM internal architecture.

At time t, x is the input variable; h is the output variable. The network uses *tanh* and sigma (σ) as activation functions; *tanh* is the hyperbolic tangent function, and σ is the sigmoid function. Their purpose is to induce nonlinear transformations in neural networks, therefore enabling higher nonlinear expression capability of the network. First, x is loaded into the network concurrently with the output data from past times. The long-term memory state variables are then selectively remembered via the forget gate, and a new memory state variable results from assuming the present state with the long-term state at the prior time over an input gate. Finally, the output variable at time t is derived from the long-term memory state variable passing via the output gate, according to Eqs. (2) to (7):

$$f_t = \sigma\left(\mathrm{Y}_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{2}$$

$$i_t = \sigma(Y_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

$$\widetilde{C}_t = \tanh(Y_C \cdot [h_{t-1}, x_t] + b_C) \tag{4}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t \tag{5}$$

$$o_t = \sigma(Y_o \cdot [h_{t-1}, x_t] + b_o) \tag{6}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{7}$$

where:

- x_t is the input at time t,
- h_t is the hidden state at time t,
- C_t is the cell state at time t,

- Y_f, Y_i, Y_C, Y_o are matrix of weights,
- b_f, b_i, b_C ,b_o represent the bias,
- σ and tanh are the sigmoid and tangent activation functions.

In this work, the LSTM is utilized to group sequential windows of the input data into features. The extracted features are fed to the XGBoost algorithm for classification, as depicted in Algorithm (1). For the input dataset (D), the feature vector Fn is extracted. In step 1, the LSTM model is defined and initialized with window size = 5 and 5 features. Each LSTM unit is represented by *num_lstm_unit* function. In step 2, the LSTM training is incorporated by selecting a window size Ws and passing it through the functionality of LSTM. Each window is computed using the forget gate (ft), gate input activation (it), candidate cell state (Ĉt), update cell state (Ct), output gate activation (ot), and hidden state (ht). After that, the model's weights are updated according to the loss function. In the last step, the features are extracted by passing each sample through the LSTM layer and extracting the hidden state (ht) as the feature vector Fn. At the end, the extracted features are stored and passed to XGBoost for classification.

Algorithm 1. LSTM Feature Extraction

```
Input: Time-series data ($D$).
Output: Extracted features ($F_h$).
Step 1: Define LSTM model
LSTM initialization
    input ($Ws$ = 5, $F_n$ = 5)
    num_lstm_unit
Step 2: LSTM training
for each epoch
    for each $Ws$:
    Compute($f_t$)
    Compute($i_t$)
    Compute ($\hat{C}_t$)
    Update ($C_t$)
    Compute($o_t$)
    Compute($h_t$)
    loss = Compute_loss(predictions, true_labels)
    Update($Y$)
Step 3: Extract Features
for each sample in $D$
    LSTM (sample)
    Extract ($h_t$)
    Store ($h_t$)
```

E) Data Sources

The proposed design is validated using the real time smart grid dataset, which is adopted from [30]. This dataset includes more than 50,000 data samples collected through 15 minutes time interval periods. The dataset contains factors and a wide range of electrical and environmental parameters such as voltage, current, humidity, active power, reactive power, temperate, and price. Portion of the dataset is illustrated in Table I below. The table shows typical values for normal behaviors and an example of abnormal behavior. The abnormal behavior is indicated by a sharp spike in current and power comparing with the surrounding time stamp, which in turn flags this time stamp as anomalous.

Table I. Portion of smart grid real-time load monitoring dataset.

Timestamp	Voltage [V]	Current [A]	Power [kW]	Reactive Power [kVAR]	Anomaly Flag
3/1/2017 0:00	230.4	5.2	1.2	0.1	0
3/1/2017 0:15	231	5.3	1.22	0.12	0
3/1/2017 0:30	229.8	48.5	11	4.3	1
3/1/2017 0:45	230.2	5.1	1.18	0.11	0

4. RESULTS AND DISCUSSION

This section presents the outcomes of the suggested model implementation, evaluated by performance measures. The following system requirements were utilized for carrying out the experiments: Intel Core i7 3.5 GHz processor, 32GB RAM, and PyCharm as the integrated development environment for the Python programming language. The explanations of the simulation's results are demonstrated in the subsequent subsections. To ensure clear representation, we divided the results into subsections according to the corresponding result type.

To test the proposed design, the "Smart Grid Real-Time Load Monitoring Dataset" was used. The dataset contains the key electrical parameters, renewable energy sources, environmental factors, and anomaly indicators that make it suitable for machine learning and deep learning optimization. The dataset properties are listed in Table 2. As described by the author of the dataset, the dataset can be used for several use cases; one of them is fault and anomaly detection [30].

Table 2. Dataset properties.

Dataset Name	Records No.	Parameters	Environmental Factors	Target Variable
Smart Grid Real-Time Load Monitoring Dataset	50,000+ records with 15-minute intervals	Voltage, current, power consumption, reactive power	Temperature, humidity, electricity price fluctuations	Predicted Load [kW]

A) Performance Evaluation

The effectiveness of the proposed approach was evaluated utilizing assessment and error measures. Accuracy, F1-score, recall, and precision serve as assessment criteria, whereas Root Mean Square Error (RMSE) and Mean Square Error (MSE) function as performance error measures. The accuracy of the system is calculated using Eq. (8)

$$\mathcal{A} = \frac{C_+ + C_-}{C_+ + C_- + I_+ + I_-} \tag{8}$$

where, $C_+ = Correct\ positive\ predictions$, $\mathrm{C}_- = Correct\ negative\ predictions$, $I_+ = Incorrect\ positive\ predictions, I_- = Incorrect\ negative\ predictions.$

The precision is calculated using Eq. (9)

$$\mathcal{P} = \frac{C_+}{C_+ + I_+} \tag{9}$$

The true positive rate or recall, is calculated using Eq. (10)

$$\mathcal{D} = \frac{C_+}{C_+ + I_-} \tag{10}$$

The harmonic performance measure or F1-score, is calculated using Eq. (11)

$$\mathcal{H} = \frac{2\mathcal{P}\mathcal{D}}{\mathcal{P} + \mathcal{D}} = \frac{2C_+}{2C_+ + I_+ + I_-} \tag{11}$$

The Quadratic Prediction Deviation (QPD), also called the Mean Square Error, is calculated using Eq. (12)

$$\phi = \frac{1}{\nu}\sum_{k=1}^{\nu}(\alpha_k - \varepsilon_k)^2 \tag{12}$$

The Root Quadratic Assessment (RQA) or the Root Mean Square Error (RMSE) is calculated using Eq. (13)

$$\varrho = \sqrt{\phi} = \sqrt{\frac{1}{\nu}\sum_{k=1}^{\nu}(\alpha_k - \varepsilon_k)^2} \tag{13}$$

For both equations (Eqs. (12) and (13)), α_k represents the values, ε_k is the estimated values, and ν is the total number of data samples.

B) Comparative Analysis

The training and validation losses of the proposed design are presented in Fig. 4. The training and validation losses were decreasing significantly with each training epoch, approaching zero. These values ensure that the proposed design is efficient and dependable. For the system parameters, we specified the number of epochs = 50, but we accomplished near-zero values (training and validation losses) before reaching this number.

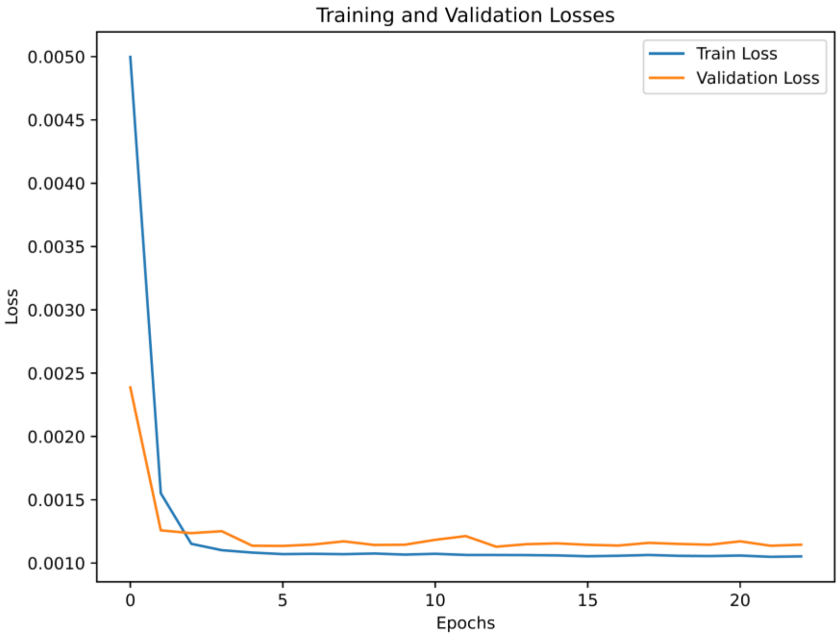

Fig. 4. The training loss and validation loss with respect to the number of epochs.

The extracted features from the LSTM stage were fed to XGBoost for classification. Fig. 5 represents the analysis of the prediction components after the XGBoost classification step. confusion matrix after the XGBoost classification. Fig. 5-a shows the confusion matrix, which represents the ratios between the different prediction scenarios. The x-axis is the projection of the predicted instances, while the y-axis represents the actual state of each instance. The true positive instances (8895) represent a large number compared to false positive instances (349). The values of the confusion matrix are used to calculate accuracy, precision, F1-score, and recall performance metrics. Fig. 5-b shows the number of normal and anomalous traffic that has been classified using XGBoost.

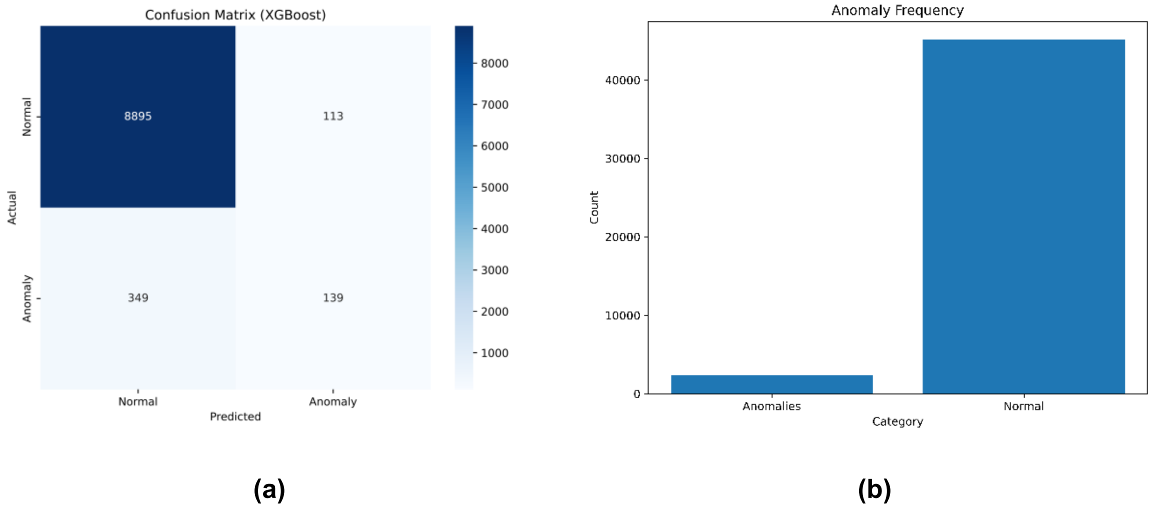

Fig. 5. Analysis of the proposed design after the XGBoost classification Step: a) Confusion Matrix; b) Number of analogous traffic.

The proposed design is compared with other works from the literature as depicted in Table 3. For the reported results, the other works reported no values for the accuracy, MSE, and RMSE. These missing values from other works show high accuracy with small mean error over time. With regard to precision, the proposed design showed a comparable result with the best work. However, precision alone does not ensure the full system effectiveness due to lack of interpretation of false negative outcomes. Therefore, the other metrics, recall and F1-score, were used to show the full performance of the system regarding all possible outcomes. The proposed design achieved the highest figures regarding recall and F1-score metrics, which indicate superiority over other works. Other work reported only accuracy, as in [13], the reported value of 95% represents a high figure in the anomaly detection field. The proposed design performed better regarding the accuracy metric. Works like [26] reported results regrading hybrid models, achieving 18.55% by combining CNN, LSTM and attention layer. The work in [27] reported 95% accuracy on average for three types of frauds classified by the design. The work in [27] didn't report the MSE and RMSE, however we calculated these values using the information they provided regarding accuracy, precision, true and false rates for positive and negative measures. Other works, like [13, 23, 28] didn't report any results or supporting measures regarding MSE or RMSE.

Table 3. Comparison of the proposed design with other work from the literature.

Work	Accuracy [%]	Precision [%]	F1-Score [%]	Recall [%]	MSE	RMSE
[13]	95	-	-	-	-	-
[23]	-	78.30	78.70	79.00	-	-
[25]	-	77.70	78.40	79.10	0.076	-
[26]	-	89.60	93.50	97.80	0.18	
[27]	95	98.60	71.60	56.30	0.051	0.226
[28]	-	99.20	94.60	98.40	-	-
Proposed	96.50	97.75	97.00	100.00	0.001	0.032

The behavior of the hybrid framework (LSTM and XGBoost) can be depicted in Fig. 6. Measurement The timeline represents the period between 2016-01 and 2017-05, with the X-axis representing time (2-month intervals) and the Y-axis representing energy consumption in kWh. As mentioned before, LSTM provides temporal feature extraction using slicing window over time series data. The XGBoost unitizes the historical output characteristic vector (h_t)

extracted from the LSTM stage. Over the selected period of time, the framework predicted anomalous action due to sudden spikes in kWh usage, e.g., 03-2017. The XGBoost marks this value as anomalous because its predicted value is much lower than the actual value from the historical data log.

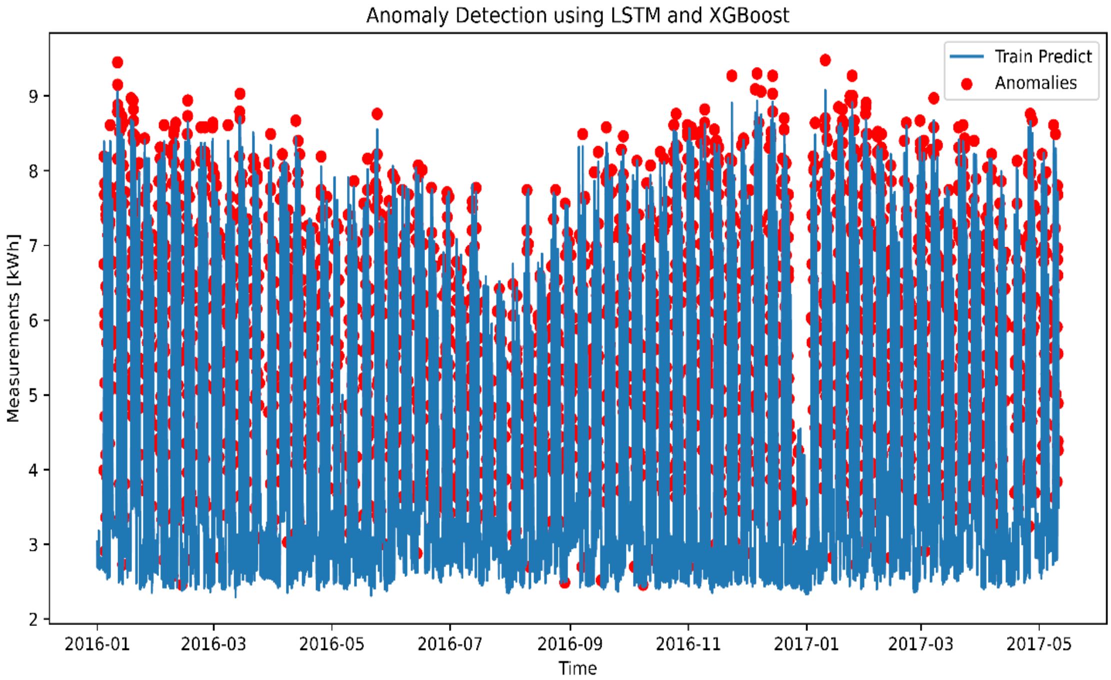

Fig. 6. Anomaly detection behavior using LSTM and XGBoost over time period for target measurements of KWh.

5. CONCLUSIONS

This work presented an anomaly detection framework for smart grids using machine learning algorithms. In the work, the long-short term memory (LSTM) and extreme gradient boosting algorithms (XGBoost) were employed. The LSTM is used to extract the system feature out of the time series dataset. The extraction utilized a proper preprocessing step, interpolation, and a sliding window for historical behavior characteristic extraction. The extracted feature's vector was fed to XGBoost for classification. The system was validated through different performance metrics and error measurements; these include accuracy, precision, F1-score, recall, MSE, and RMSE. The reported results showed the significance of the proposed design in detecting anomalous actions out of normal system behavior. In the future, more experiments will be conducted using different types and sizes of datasets utilizing different deep learning algorithms and auto-encoders. The experiments will focus on the hyperparameter setting of deep learning and classification algorithms.

There are a number of limitations that are worth mentioning to consider and overcome in future research. First, the time complexity of LSTM and XGBoost training, validation, and classifications. Second, the sensitivity of hyperparameters, which requires extensive tuning. Lastly, the performance degradation for the large-scale grids or heterogeneous datasets.

Acknowledgment

The authors would like to acknowledge the use of the Smart Grid Real-Time Load Monitoring Dataset available on Kaggle [26].

REFERENCES

[1] L. Tailhardat, Y. Chabot, R. Troncy, "NORIA-O: an ontology for anomaly detection and incident management in ICT systems," European Semantic Web Conference, 2024, doi: 10.1007/978-3-031-60635-9_2.

[2] H. Choi, J. Kim, "Anomaly detection system of smart farm ICT device", *The Institute of Internet, Broadcasting and Communication*, vol. 19, no. 2, pp. 169-174, 2019.

[3] H. AlEisa, F. Alrowais, R. Allafi, N. Almalki, R. Faqih, R. Marzouk, "Transforming transportation: Safe and secure vehicular communication and anomaly detection with intelligent cyber--physical system and deep learning," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 1736–1746, 2023, doi: 10.1109/TCE.2023.3325827.

[4] M. Hassan, M. Rehmani, J. Chen, "Anomaly detection in blockchain networks: a comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 289–318, 2022, doi: 10.1109/COMST.2022.3205643.

[5] W. Lim, K. Yong, B. Lau, C. Tan, "Future of generative adversarial networks (GAN) for anomaly detection in network security: A review," *Computers & Security*, vol. 139, p. 103733, 2024, doi: 10.1016/j.cose.2024.103733.

[6] J. McCarthy, M. Powell, K. Stouffer, C. Tang, T. Zimmerman, W. Barker, T. Ogunyale, D. Wynne, J. Wiltberger, *Securing Manufacturing Industrial Control Systems: Behavioral Anomaly Detection*, US Department of Commerce, National Institute of Standards and Technology, 2020.

[7] E. Al-qtiemat and Z. Al-odat, "Examining cloud security: identifying risks and the implemented mitigation strategies," *Journal of Theoretical and Applied Information Technology*, vol. 102, no. 7, 2024.

[8] B. Khan, H. Getachew, H. Alhelou, "Components of the smart-grid system," *Solving Urban Infrastructure Problems Using Smart City Technologies*, 2021, pp. 385–397, doi: 10.1016/B978-0-12-816816-5.00017-6.

[9] A. Chatterjee, B. Ahmed, "IoT anomaly detection methods and applications: a survey," *Internet of Things*, vol. 19, p. 100568, 2022, doi: 10.1016/j.iot.2022.100568.

[10] M. Hossain, A. Oo, A. Ali, *Smart Grid, Opportunities, Developments, And Trends*, Springer, 2013.

[11] M. Douiba, S. Benkirane, A. Guezzaz, M. Azrour, "An improved anomaly detection model for IoT security using decision tree and gradient boosting," *The Journal of Supercomputing*, vol. 79, no. 3, pp. 3392–3411, 2023, doi: 10.1007/s11227-022-04783-y.

[12] A. Zemliak, A. Osadchuk, "Analysis of the circuit optimization process based on a generalized approach and a genetic algorithm," *Jordan Journal of Electrical Engineering*, vol. 10, no. 1, pp. 1-26, 2024, doi: 10.5455/jjee.204-1679101785, doi: 10.5455/jjee.204-1679101785.

[13] A. Abusitta, G. Carvalho, O. Wahab, T. Halabi, B. Fung, S. Al Mamoori, "Deep learning-enabled anomaly detection for IoT systems," *Internet of Things*, vol. 21, p. 100656, 2023, doi: 10.1016/j.iot.2022.100656.

[14] P. Shabad, A. Alrashide, O. Mohammed, "Anomaly detection in smart grids using machine learning," *Annual Conference of the IEEE Industrial Electronics Society*, 2021, doi: 10.1109/IECON48115.2021.9589851.

[15] M. Mohamed, M. Bilal, "Comparing the performance of deep denoising sparse autoencoder with other defense methods against adversarial attacks for arabic letters," *Jordan Journal of Electrical Engineering*, vol. 10, no. 1, pp. 122-133, 2024, doi: 10.5455/jjee.204-1687363297.

[16] M. Hooshmand, D. Hosahalli, "Network anomaly detection using deep learning techniques," *CAAI Transactions on Intelligence Technology*, vol. 7, no. 2, pp. 228–243, 2022, doi: 10.1049/cit2.12078.

[17] A. Muppidi, H. Nannam, "Experimental evaluation of a proposed LQR-LU optimal grid controller in the applications of grid-tied PV systems," *Jordan Journal of Electrical Engineering*, vol. 11, no. 1, pp. 100-111, 2024, doi: 10.5455/jjee.204-1715856762.

[18] M. Bartouli, I. Hagui, A. Msolli, A. Helali, F. Hassen, "Smart grid load forecasting models using recurrent neural network and Long Short-Term memory," *Jordan Journal of Electrical Engineering*,

vol. 11, no. 1, pp. 35-51, 2024, doi: 10.5455/jjee.204-1703066445.

[19] M. Hasan, R. Toma, A. Nahid, M. Islam, J. Kim, "Electricity theft detection in smart grid systems: a CNN-LSTM based approach," *Energies*, vol. 12, no. 17, 2019, doi: 10.3390/en12173310.

[20] Y. Peng, Y. Yang, Y. Xu, Y. Xue, R. Song, J. Kang,, "Electricity theft detection in AMI based on clustering and local outlier factor," *IEEE Access*, vol. 9, pp. 107250–107259, 2021, doi: 10.1109/ACCESS.2021.3100980.

[21] A. Almazroi, N. Ayub, "A novel method CNN-LSTM ensembler based on black widow and blue monkey optimizer for electricity theft detection," *IEEE Access*, vol. 9, pp. 141154–141166, 2021, doi: 10.5455/jjee.204-1703066445.

[22] M. Adil, N. Javaid, U. Qasim, I. Ullah, M. Shafiq, J. Choi, "LSTM and BAT-based RUSBoost approach for electricity theft detection," *Applied Sciences*, vol. 10, no. 12, p. 4378, 2020, doi: 10.3390/app10124378.

[23] S. Ali, G. Wang, R. Cottrell, T. Anwar, "Detecting anomalies from end-to-end internet performance measurements (PingER) using cluster based local outlier factor," IEEE international symposium on parallel and distributed processing with applications and IEEE international conference on ubiquitous computing and communication, 2017, doi: 10.1109/ISPA/IUCC.2017.00150.

[24] S. Munawar, M. Asif, B. Kabir, Pamir, A. Ullah, N. Javaid, "Electricity theft detection in smart meters using a hybrid Bi-directional GRU Bi-directional LSTM model," Complex, Intelligent and Software Intensive Systems, 2021, pp. 297–308, doi: 10.1007/978-3-030-79725-6_29.

[25] K. Kammerer, B. Hoppenstedt, R. Pryss, S. Stökler, J. Allgaier, M. Reichert, "Anomaly detections for manufacturing systems based on sensor data—insights into two challenging real-world production settings," *Sensors*, vol. 19, no. 24, p. 5370, 2019, doi: 10.3390/s19245370.

[26] J. Kong, W. Jiang, Q. Tian, M. Jiang, T. Liu, "Anomaly detection based on joint spatio-temporal learning for building electricity consumption," *Applied Energy*, vol. 334, p. 120635, 2023, doi: 10.1016/j.apenergy.2022.120635.

[27] A. korba, N. karabadji, "Smart grid energy fraud detection using SVM," I*nt*ernational Conference On Networking And Advanced Systems, 2019, doi: 10.1109/ICNAS.2019.8807832.

[28] S. Evangeline, S. Darwin, P. Anandkumar, M. Thanu, "Anomaly detection in smart grid using a trace-based graph deep learning model," *Electrical Engineering*, vol. 106, no. 5, pp. 5851–5867, 2024, doi: 10.1007/s00202-024-02327-6.

[29] T. Chen, C. Guestrin, "XGBoost: a scalable tree boosting system," 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, doi: 10.1145/2939672.2939785.

[30] *Smart Grid Real-Time Load Monitoring Dataset*, 2025, https://www.kaggle.com/datasets/ziya07/smart-grid-real-time-load-monitoring-dataset.