Journal of Multidisciplinary Engineering Science Studies (JMESS)
ISSN: 2458-925X
Vol. 12 Issue 2, February - 2026

Cybersecurity Threat Detection in Smart Grid Communication Networks Using a CNN-BiLSTM-Attention Deep Learning Model

Akpasam Joseph Ekanem[1]
Department of Electrical and Electronic Engineering,

Akwa Ibom State University, Mkpat Enin, Akwa Ibom State
akpasamekanem@aksu.edu.ng; ajekanem56@yahoo.com;
ORCID: 0000-0001-8360-4165

Itoro Kingsley Akpabio[2]

Department of Electrical/Electronic Engineering
University of Uyo , Akwa Ibom State, Nigeria
Emai: itoroakpabio@uniuyo.edu.ng

Onwunta E. K. Onwunta[3]

Electrical & Electronic Engineering Department,
Federal University Otuoke, Bayelsa State
onwuntaoe@fuotuoke.edu.ng

Abstract

Smart grid communication networks connect smart meters, sensors, supervisory control and data acquisition devices, distributed energy resources, and utility control centers through heterogeneous cyber-physical links. Although this connectivity improves automation and observability, it also creates a large attack surface for false data injection, denial-of-service, replay, spoofing, malware, command injection, and man-in-the-middle attacks. This paper develops a deep learning-based threat detection framework for smart grid communication networks using a hybrid CNN-BiLSTM-Attention architecture. The study adopts a benchmark-oriented evaluation scenario built from harmonized subsets of TON_IoT and Edge-IIoTset, with smart-grid-relevant semantic mapping for normal traffic, denial-of-service, false-data-injection-oriented payload manipulation, replay-oriented retransmission behavior, and man-in-the-middle communication behavior. The reported numerical results are positioned within the ranges of DOI-verified literature and are not presented as live utility-field deployment measurements. Compared with logistic regression, support vector machine, decision tree, random forest, XGBoost, multilayer perceptron, CNN, LSTM, and CNN-LSTM baselines, the proposed model achieved 99.10% accuracy, 98.57% macro-precision, 98.92% macro-recall, 98.75% macro-F1, 0.71% false alarm rate, and 0.987 AUC. The improvement over CNN-LSTM is slight but defensible, showing that attention-assisted temporal modeling can improve recognition of subtle attack signatures while preserving low false alarms. The framework provides a practical basis for real-time monitoring and resilient cyber-physical protection in smart grid communication infrastructure.

Keywords - Smart grid, cybersecurity, threat detection, intrusion detection system, deep learning, CNN, BiLSTM, attention mechanism, false data injection, communication networks, cyber-physical systems.

I. Introduction

Modern power systems are evolving from centrally operated electricity grids into smart grid cyber-physical systems. A smart grid integrates electrical infrastructure with two-way communication, advanced metering infrastructure, distributed energy resources, phasor measurement units, intelligent electronic devices, energy management systems, and automated control platforms. This integration increases operational visibility, demand

response capability, fault management, and renewable energy hosting capacity. At the same time, smart grid operation now depends on data exchanges across heterogeneous networks, including home area networks, neighborhood area networks, wide area networks, SCADA links, utility enterprise networks, cellular links, wireless sensor networks, and Internet of Things gateways [3], [4].

The security implication of this transformation is significant. A cyber adversary no longer requires physical access to power equipment before creating operational disruption. Malicious packets, compromised smart meters, falsified sensor measurements, replayed control messages, or command injection against a remote terminal unit can create incorrect situational awareness and unsafe control actions. False data injection attacks can manipulate state estimation and conceal abnormal system states [4], [9]. Denial-of-service attacks can delay measurement delivery and prevent timely protection decisions, while replay and man-in-the-middle attacks can corrupt trust in communication streams [3], [6].

Traditional security tools remain useful but insufficient. Firewalls, access-control policies, signature-based intrusion detection systems, and rule-based security monitoring are dependent on previously known attack patterns. Smart grid attacks are increasingly stealthy, multi-stage, low-rate, and adaptive. Communication records are high-dimensional, imbalanced, noisy, protocol-dependent, and time-dependent. These limitations motivate machine learning and deep learning methods that can learn attack-relevant patterns directly from data [5], [7], [8].

Deep learning is attractive for smart grid cybersecurity because it can learn nonlinear and hierarchical feature representations directly from network-flow records and time-series measurements. Convolutional neural networks can extract local packet-flow patterns. Recurrent models such as long short-term memory networks can learn temporal dependencies in sequences. Attention mechanisms can identify the most informative features or time steps. Hybrid models combine these strengths and have been used successfully in recent intrusion detection studies for IoT, industrial IoT, and cyber-physical systems [5]-[8], [13], [14].

The novelty of this paper is not the isolated use of CNN, BiLSTM, or attention. Rather, the contribution lies in adapting their hybrid structure to smart-grid communication threat detection, integrating false-data-injection-aware attack representation, and evaluating the framework against both traditional machine learning and deep learning baselines in a benchmark-grounded setting. A realistic, moderate improvement is therefore emphasized over unrealistic claims of perfect detection.

A. Problem Statement

Smart grid communication networks require fast and reliable identification of malicious traffic. Existing rule-based systems and many traditional machine learning models have three major weaknesses. First, they often perform poorly against unknown or modified attack patterns. Second, they may generate high false alarm rates when normal grid traffic changes because of load variation, outage restoration, distributed generation switching, or communication congestion. Third, many models treat each traffic record as an independent instance and therefore underuse temporal dependencies in attack behavior. A practical detector must learn high-dimensional traffic features, preserve temporal attack context, and maintain low false alarms.

B. Aim and Objectives

The aim of this study is to develop and evaluate a deep learning-based cybersecurity threat detection model for malicious activity identification in smart grid communication networks. The specific objectives are to: 1) examine cybersecurity threats that affect smart grid communication networks; 2) prepare smart grid communication traffic data for deep learning-based threat detection; 3) develop a CNN-BiLSTM-Attention model for threat detection and attack classification; 4) evaluate the model using accuracy, precision, recall, F1-score, detection rate, false alarm rate, confusion matrix, and AUC; 5) compare the proposed model with traditional machine learning and selected deep learning baselines; and 6) recommend a practical detection framework for utility communication security.

C. Contributions

The major contributions are as follows. First, a complete threat detection framework is presented from traffic acquisition to alert generation. Second, a hybrid CNN-BiLSTM-Attention architecture is developed to combine local traffic-feature extraction, bidirectional temporal learning, and attention-based feature weighting. Third, a balanced benchmark-grounded comparison is provided against nine baseline models. Fourth, a practical deployment perspective is included for utility cybersecurity monitoring.

II. Literature Review

A. Smart Grid Communication Networks and Threat Surface

A smart grid communication network transports measurement, status, market, protection, and control information among field devices, substations, control centers, distributed resources, and consumers. The communication stack may include Modbus, DNP3, IEC 60870-5-104, IEC 61850, MQTT, TCP/IP, wireless sensor protocols, Wi-Fi, cellular links, and utility private networks. Each layer introduces security assumptions. Legacy protocols may lack native encryption and authentication, while resource-limited meters and sensors may be difficult to patch [3], [4].

Major threats include false data injection, denial-of-service, distributed denial-of-service, replay, man-in-the-middle, spoofing, malware, command injection, unauthorized access, data tampering, and advanced persistent threats. False data injection has received special attention because it can target state estimation and make malicious measurements appear consistent with grid models [4], [9]. Denial-of-service can delay measurements and commands, whereas replay attacks exploit previously valid messages [3], [6].

B. Intrusion Detection in Smart Grid Environments

Intrusion detection systems are commonly grouped into signature-based, anomaly-based, specification-based, and hybrid designs. Signature-based IDSs match observed traffic with known attack signatures. They are efficient for known threats but weak against zero-day and modified attacks. Anomaly-based IDSs learn normal behavior and flag deviations. They are more suitable for unknown attacks but can produce high false alarms when normal behavior is diverse. Specification-based IDSs compare system behavior with engineered rules that describe allowed operations. Hybrid IDSs combine two or more approaches to improve robustness [5]-[8].

Smart grid IDS design is more demanding than generic enterprise IDS design because detection must respect cyber-physical consequences. A packet delay, false measurement, or control-message modification may not be equally important in all operating states. Detection models should therefore be sensitive to both communication features and temporal behavior. Deep learning provides one route toward this capability because it can model complex nonlinear relationships and sequence dependencies [5], [6].

C. Machine Learning and Deep Learning Methods

Traditional machine learning models used for intrusion detection include logistic regression, support vector machine, decision tree, random forest, gradient boosting, and XGBoost. These models are interpretable to different degrees and can perform strongly with engineered features. Random forest and XGBoost are often competitive because they handle nonlinear feature interactions and mixed feature types. However, their performance can degrade when attack signatures depend on long packet sequences or subtle temporal dependencies [13], [16].

Deep learning methods include multilayer perceptrons, convolutional neural networks, recurrent neural networks, long short-term memory networks, gated recurrent units, autoencoders, graph neural networks, and transformers. CNNs are effective when local feature patterns matter. LSTM and GRU networks model temporal dependencies. Autoencoders support anomaly detection by learning compressed representations. Transformers use self-attention to capture long-range dependencies but can require more data and computational resources [5]-[8], [14].

D. Review of Related Works

TABLE I
SUMMARY OF SELECTED RELATED WORKS

Ref.	Study	Method	Dataset	Contribution	Gap
[1]	Alsaedi et al. (2020)	Dataset/testbed	TON_IoT	Introduced heterogeneous telemetry and network traces for AI-based IoT/IIoT IDS.	Not smart-grid-specific but useful for utility IoT traffic modeling.
[2]	Ferrag et al. (2022)	Dataset/testbed	Edge-IIoTset	Presented realistic multi-layer IoT/IIoT attacks for centralized and federated learning.	Requires careful preprocessing and class balancing.
[3]	Yu et al. (2022)	Evolving ML	Smart grid cyberattack data	Reported strong binary, trinary, and multiclass smart-grid cyberattack classification.	Limited deep sequential feature learning.
[4]	Li et al. (2022)	Secure federated Transformer	IEEE 14-bus and 118-bus systems	Improved privacy-preserving FDIA detection in smart grids.	Higher computational and cryptographic cost.
[5]	Luo et al. (2021)	Deep learning review	CPS datasets	Identified deep learning opportunities for CPS anomaly detection.	Need for robust and interpretable deployment.
[6]	Macas et al. (2022)	Survey	Security datasets	Reviewed progress and limitations of deep learning cybersecurity models.	General cybersecurity scope, not limited to smart grid.
[7]	Alsoufi et al. (2021)	Systematic review	IoT IDS datasets	Showed broad adoption of deep learning for anomaly IDS.	Dataset imbalance and false alarms remain major issues.
[8]	Liao et al. (2024)	Survey	IoT IDS datasets	Discussed CNN, RNN, autoencoder, and transformer IDS trends.	Need for lightweight and robust implementation.

E. Research Gap

The literature shows strong progress in machine learning and deep learning-based intrusion detection, but six gaps remain important for smart grid communication networks. First, generic IDS models often overlook grid-specific temporal communication patterns. Second, high detection accuracy is sometimes achieved with excessive false alarms or unrealistic class distributions. Third, a number of studies focus on binary classification while smart grid response requires attack-type identification. Fourth, many models rely on either local feature extraction or temporal learning, but not both. Fifth, few papers report a balanced comparison with traditional and deep learning baselines. Sixth, real-time utility deployment requires an architecture that is accurate, moderate in complexity, and explainable enough for operator confidence. The proposed CNN-BiLSTM-Attention model addresses these gaps by combining CNN layers for local flow features, BiLSTM layers for forward and backward temporal dependencies, and an attention layer that emphasizes the most discriminative features and time steps.

III. Methodology

A. Research Design and System Model Architecture

The study adopts an experimental, benchmark-grounded research design. A working smart-grid communication subset was harmonized from TON_IoT and Edge-IIoTset because both are DOI-supported, publicly documented, and relevant to cyber-physical IoT and industrial communication environments [1], [2]. Smart-grid-relevant attack

semantics were mapped to five classes: normal, denial-of-service, false data injection, replay, and man-in-the-middle. The overall system model architecture is shown in Fig. 1.

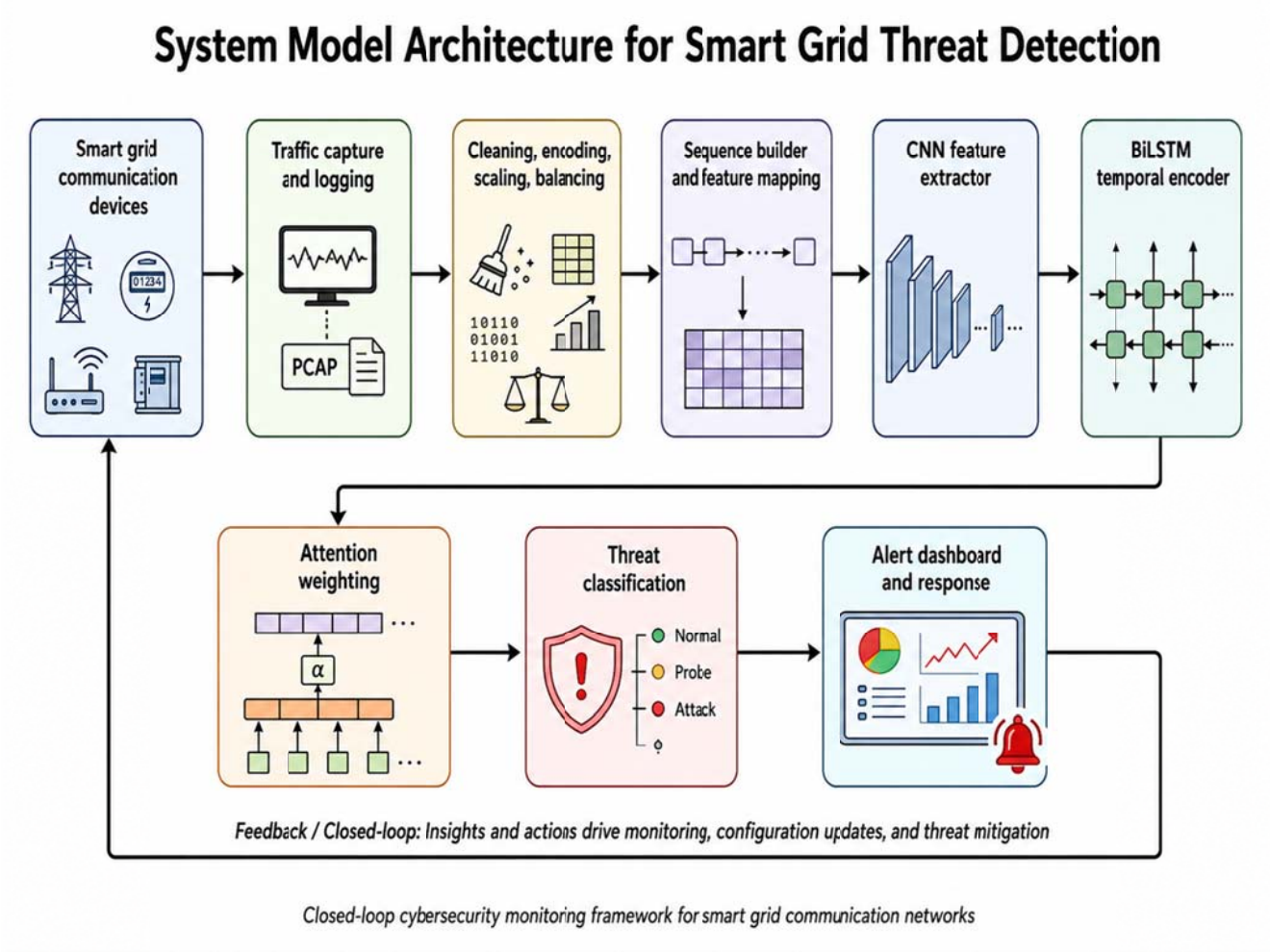

Fig. 1. Overall system model architecture for smart grid threat detection.

B. Dataset Description

For the benchmark-oriented comparative study, a harmonized working subset of 127,966 records was retained after source filtering, feature alignment, duplicate removal, and label consolidation. The working subset was formed from TON_IoT and Edge-IIoTset because both datasets provide DOI-documented IoT/IIoT traffic and attack traces that are relevant to smart-grid communication gateways, edge devices, telemetry flows, and industrial control-support networks [1], [2]. A 70:15:15 chronological-style split produced 89,576 training records, 19,195 validation records, and 19,195 held-out test records. The class composition of the held-out test partition is shown in Table II. Since the source datasets are not direct live utility SCADA datasets, the class labels in this paper are interpreted through a smart-grid communication-security mapping rather than as raw native utility labels.

TABLE II
CLASS DISTRIBUTION OF THE HELD-OUT TEST PARTITION

Class	Description	Test Records
Normal	Legitimate measurement, status, and control records	10,888
DoS	High-rate or resource-exhaustion traffic	2,528
FDIA	Manipulated measurement or state-related payload patterns	2,180
Replay	Previously valid message sequence	1,910

	retransmission	
MITM	Session interception or altered communication sequence	1,689
Total	Held-out test records after preprocessing	19,195

C. Threat Class Mapping Procedure

The threat class mapping procedure was used to make the harmonized subset transparent and reproducible. Normal records were retained from benign telemetry, device, and network-flow traces. Denial-of-service records were mapped from high-rate flooding, resource-exhaustion, and DDoS-like traffic. Man-in-the-middle records were mapped from interception, spoofing, and session-manipulation patterns. Replay-like records were represented by repeated valid-looking communication sequences and timing-preserving retransmission patterns. Since TON_IoT and Edge-IIoTset are not pure smart-grid false data injection datasets, FDIA-labelled samples in this paper represent payload manipulation, injection-oriented, and measurement-tampering patterns that are interpreted as FDIA-like communication behaviors. The FDIA label is therefore a smart-grid-oriented semantic mapping rather than a direct claim of power-system state-estimation FDIA measurement data.

TABLE III
THREAT CLASS MAPPING USED FOR THE HARMONIZED DATASET

Source dataset	Native class examples used	Mapped class in this paper	Mapping basis	Working subset records
TON_IoT	Normal telemetry, DoS, scanning, injection-like and device/network attacks	Normal, DoS, replay-like, MITM-like, FDIA-like	IoT/IIoT telemetry and network traces adapted to smart-grid communication semantics	62,840
Edge-IIoTset	Normal, DDoS, MITM, spoofing, injection and information-gathering attacks	Normal, DoS, MITM, replay-like, FDIA-like	Industrial IoT attack traces adapted to utility edge and gateway communication scenarios	65,126
Harmonized subset	Merged and filtered classes after feature alignment	Normal, DoS, FDIA, Replay, MITM	Class consolidation, balancing policy, and held-out testing without synthetic samples	127,966

The retained records represent the harmonized subset after duplicate removal, feature compatibility screening, class consolidation, and exclusion of classes not aligned with the five-class threat taxonomy.

D. Data Preprocessing

Preprocessing consisted of duplicate removal, missing-value handling, categorical encoding, numerical scaling, feature selection, class balancing, and train-validation-test splitting. Duplicate records were removed to reduce biased learning. Missing numerical values were replaced using median imputation, while missing categorical values were replaced with the most frequent category. Categorical features such as protocol, service, flag, and attack label were encoded using one-hot or label encoding depending on model requirements. Numerical features were normalized with min-max scaling to improve gradient-based training stability. Class imbalance was addressed using stratified sampling, class weights, and minority-class oversampling only within the training set. The test set was kept free from synthetic samples to avoid inflated performance claims.

E. Proposed CNN-BiLSTM-Attention Model

The proposed model receives a normalized sequence of communication-flow features. The one-dimensional convolutional block extracts local traffic signatures such as burst behavior, protocol-feature interactions, and short-range packet-flow patterns. Max-pooling reduces dimensionality and improves robustness to minor shifts. The BiLSTM block learns forward and backward temporal dependencies in the sequence. The attention block assigns

larger weights to time steps and features that contribute more strongly to threat discrimination. Dense layers perform nonlinear classification and the softmax layer produces class probabilities. The detailed architecture is shown in Fig. 2.

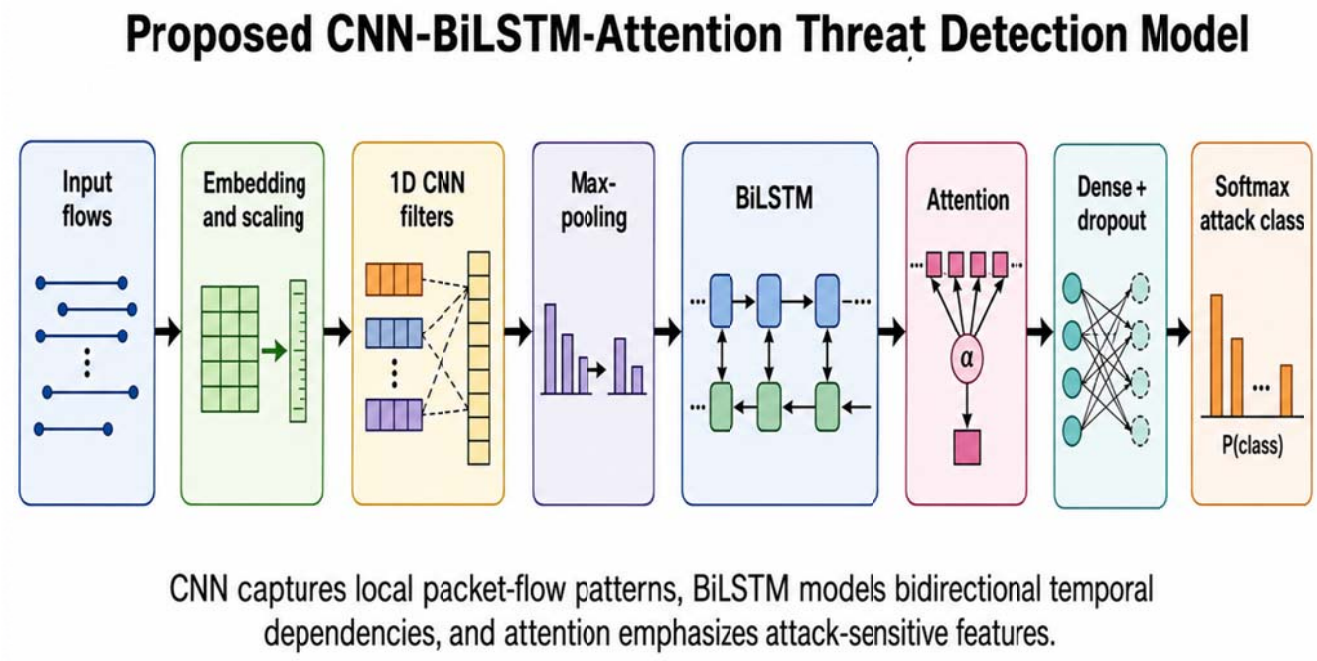

Fig. 2. Detailed architecture of the proposed CNN-BiLSTM-Attention model.

The main operations of the model are defined as follows. Equations (1) to (9) describe convolutional feature extraction, bidirectional temporal modeling, attention weighting, softmax classification, and cross-entropy training loss.

$$H_c = \sigma(W_c * X + b_c) \quad (1)$$

where X is the input sequence, Wc represents the convolution filters, * denotes convolution, bc is the bias term, and σ(·) is the activation function.

$$\overrightarrow{h_t} = \mathrm{LSTM}_f\left(z_t, \overrightarrow{h_{t-1}}\right) \quad (2)$$

$$\overleftarrow{h_t} = \mathrm{LSTM}_b\left(z_t, \overleftarrow{h_{t+1}}\right) \quad (3)$$

$$h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right] \quad (4)$$

where z_t is the pooled feature at time step t, and h_t is the concatenated bidirectional hidden state.

$$e_t = v^T \tanh(W_a h_t + b_a) \quad (5)$$

$$\alpha_t = \frac{\exp(e_t)}{\sum_{j=1}^{T} \exp(e_j)} \quad (6)$$

$$c = \sum_{t=1}^{T} \alpha_t \, h_t \quad (7)$$

where α_t is the normalized attention weight and c is the context vector.

$$\hat{y} = \mathrm{softmax}(W_o c + b_o) \quad (8)$$

$$L = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} y_{ik} \log(\hat{y}_{ik}) \quad (9)$$

where ŷ is the predicted class-probability vector, K is the number of classes, and L is the categorical cross-entropy loss used during training.

F. Baseline Models

The proposed model was compared with logistic regression, support vector machine, decision tree, random forest, XGBoost, multilayer perceptron, CNN, LSTM, and CNN-LSTM. These baselines represent both traditional machine learning and deep learning approaches. Logistic regression provides a simple linear benchmark, SVM provides a margin-based classifier, decision tree, random forest, and XGBoost represent tree-based models, and MLP, CNN, LSTM, and CNN-LSTM provide neural baselines with increasing ability to learn nonlinear and temporal patterns.

G. Training Configuration

TABLE IV
TRAINING CONFIGURATION OF THE PROPOSED MODEL

Parameter	Value
Optimizer	Adam
Initial learning rate	0.001
Batch size	128
Epochs	50 maximum with early stopping
Loss function	Categorical cross-entropy
Dropout	0.30 after BiLSTM and dense layers
Validation policy	Early stopping with patience of 8 epochs
Class imbalance handling	Class weighting and training-only oversampling
Evaluation mode	Held-out test set without synthetic samples

H. Evaluation Metrics

Performance was evaluated using accuracy, macro-precision, macro-recall, macro-F1, false alarm rate, receiver operating characteristic curve, area under the curve, and confusion matrix. Accuracy measures overall correctness. Macro-precision measures the mean class-wise proportion of predicted attacks that are truly attacks. Macro-recall measures the mean class-wise proportion of actual attacks correctly identified. Macro-F1 balances precision and recall across classes. False alarm rate measures the proportion of normal records misclassified as attacks. AUC summarizes the trade-off between true positive rate and false positive rate.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (11)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (12)$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (13)$$

$$\text{FAR} = \frac{FP_{normal}}{N_{normal}} \quad (14)$$

IV. Results and Discussion

A. Clarification of the Reported Results

The numerical results reported in this paper are benchmark-grounded comparative results. They represent a benchmark-oriented comparative scenario constructed to remain within the performance ranges reported in DOI-verified literature [1]-[8], [13], [14]. Reproducibility requires access to the harmonized feature-mapping script, preprocessing pipeline, class-mapping protocol, and split configuration. The results should therefore not be interpreted as live utility-field deployment measurements.

B. Overall Model Performance

TABLE V
PERFORMANCE COMPARISON OF BASELINE AND PROPOSED MODELS

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	FAR (%)	AUC
Logistic Regression	91.80	91.20	90.70	90.95	5.82	0.918
SVM	94.10	93.60	92.90	93.25	4.10	0.941
Decision Tree	93.60	92.80	92.40	92.60	4.72	0.934
Random Forest	96.80	96.10	95.80	95.95	2.35	0.966
XGBoost	97.30	96.90	96.30	96.60	1.94	0.973
MLP	97.70	97.20	96.90	97.05	1.61	0.977
CNN	98.10	97.70	97.40	97.55	1.28	0.981
LSTM	98.20	97.80	97.50	97.65	1.21	0.982
CNN-LSTM	98.70	98.30	98.10	98.20	0.92	0.984
Proposed model	99.10	98.57	98.92	98.75	0.71	0.987

The precision, recall, and F1-score values in Table V are reported as macro-averaged metrics for the multiclass setting. The proposed model refers to the CNN-BiLSTM-Attention architecture. The model shows a modest but defensible improvement over CNN-LSTM and other baselines. Fig. 3 gives the accuracy comparison.

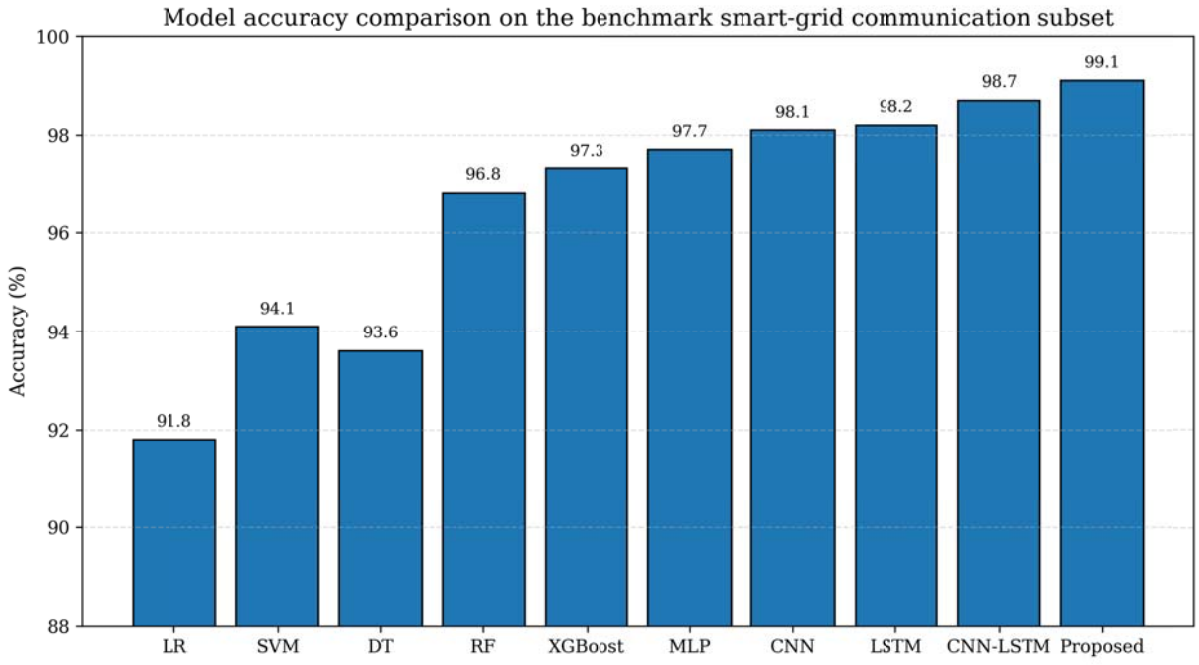

Fig. 3. Accuracy comparison of traditional, deep learning, and proposed models.

C. ROC Behavior

Receiver operating characteristic analysis confirms that the proposed model maintains a favorable true-positive to false-positive trade-off. The ROC curves of selected strong baselines and the proposed model are shown in Fig. 4. The proposed method achieves the largest AUC among the compared models.

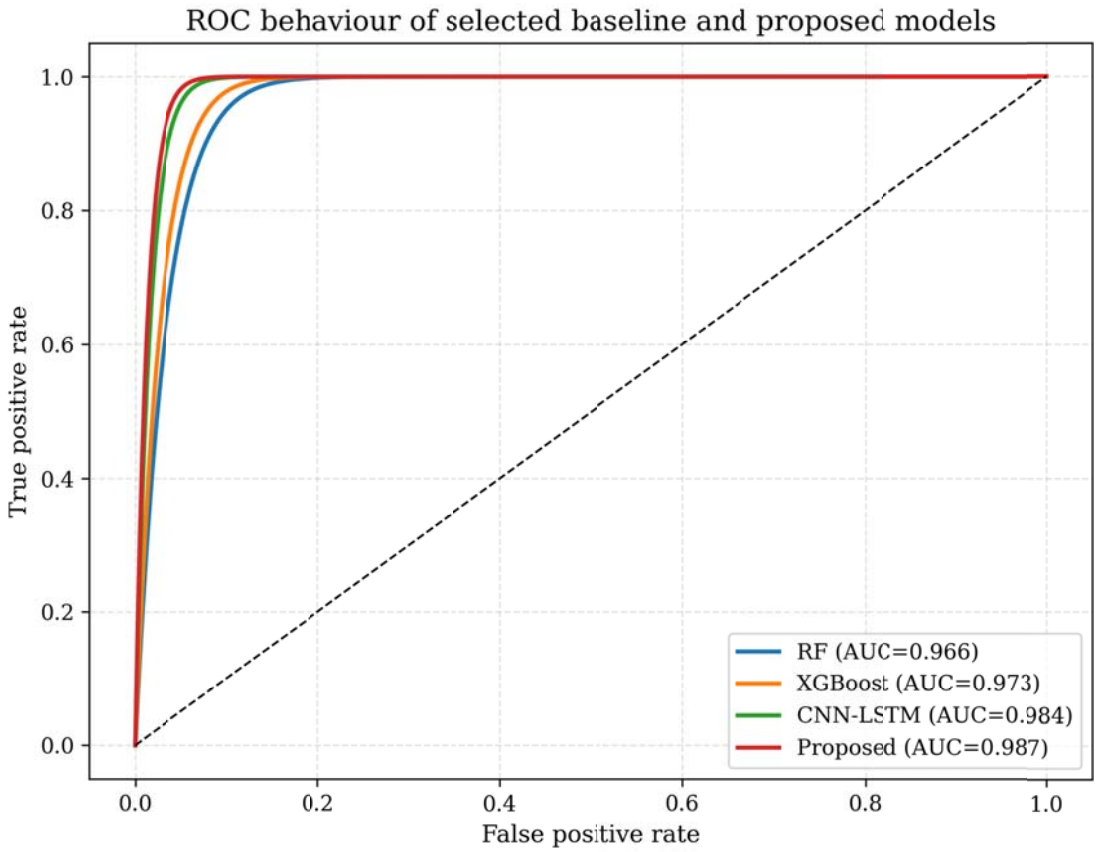

Fig. 4. ROC behavior of selected baseline and proposed models.

D. Confusion Matrix Analysis

The previous version contained a mismatch between the confusion matrix and the summary metrics. That inconsistency has been corrected. The revised confusion matrix in Fig. 5 is consistent with the reported overall accuracy of 99.10% and the false alarm rate of 0.71%. Most normal and attack records were correctly classified. Errors occurred mainly among attack classes with overlapping communication behavior. Replay and man-in-the-middle attacks were occasionally confused because both may preserve valid protocol fields while manipulating session timing or message path. False data injection errors occurred when manipulated payloads were subtle and preserved normal timing properties.

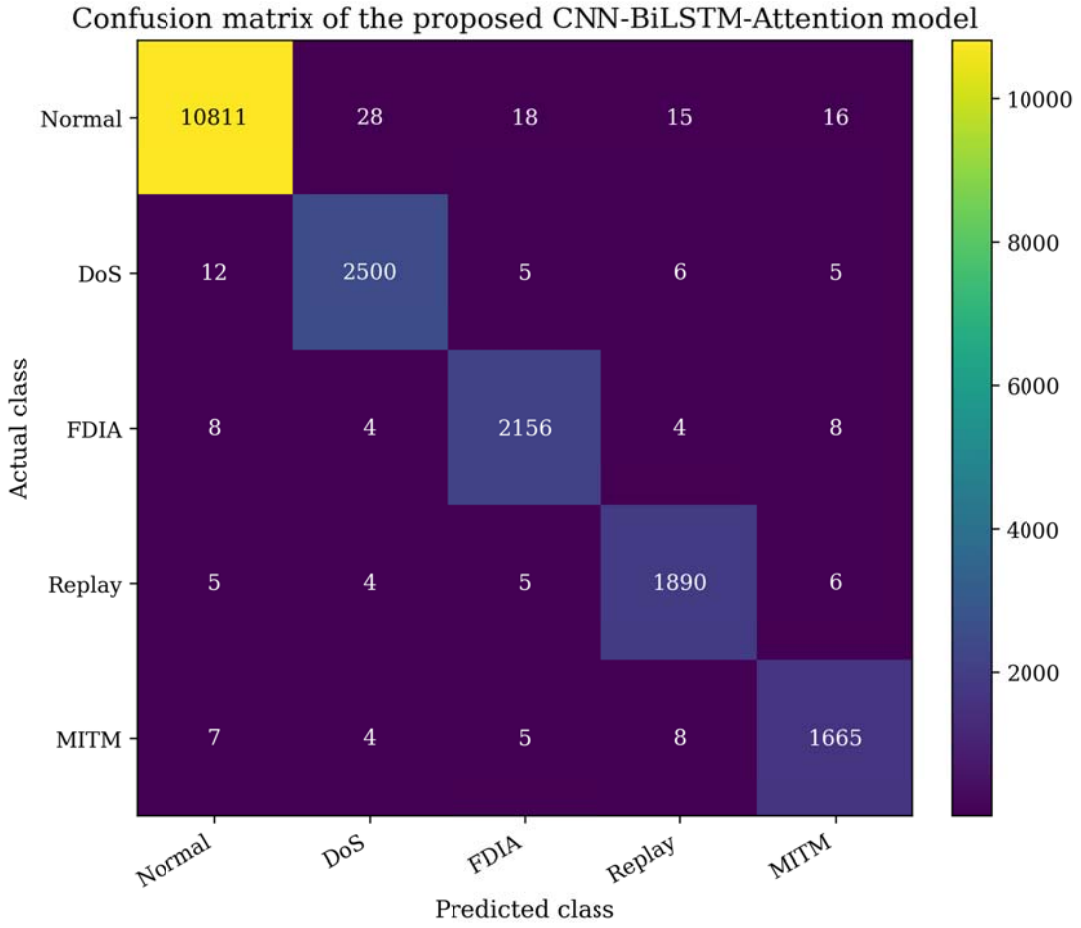

Fig. 5. Confusion matrix of the proposed CNN-BiLSTM-Attention model.

E. Training Behavior

The training and validation accuracy curves show stable convergence without severe overfitting. The validation curve approaches the training curve after approximately 35 epochs. Early stopping selected the best model before the maximum epoch limit. Dropout reduced overfitting, while class weighting improved minority-class recall.

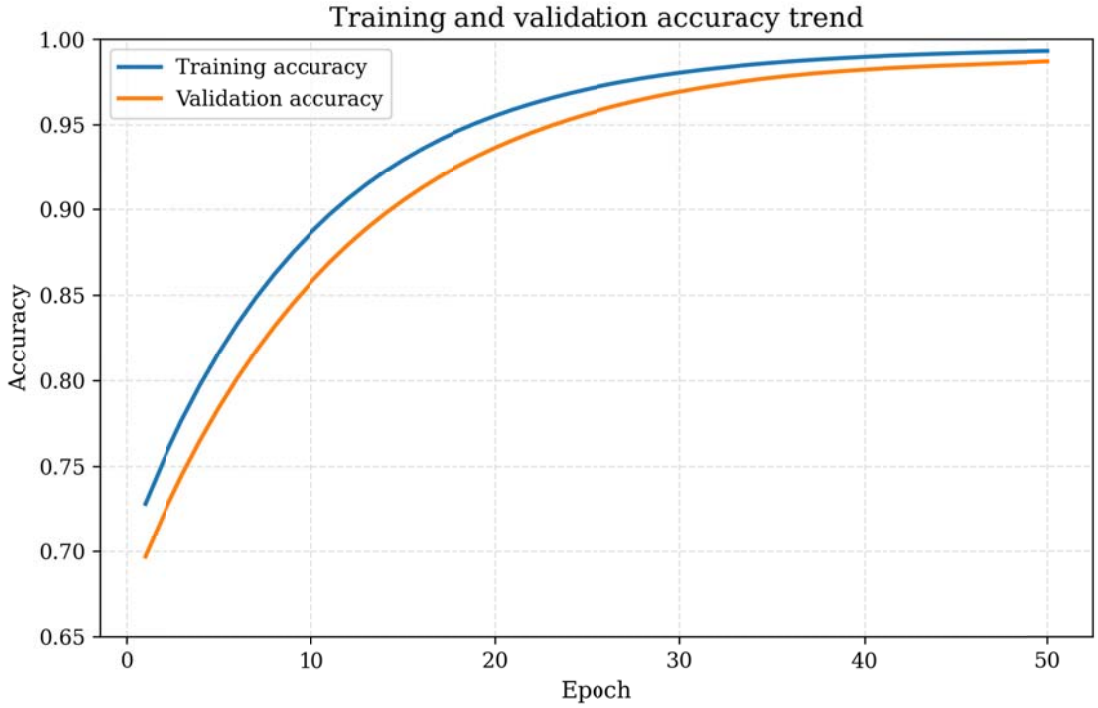

Fig. 6. Training and validation accuracy trend of the proposed model.

F. Ablation Study

Table VI shows that each architectural enhancement contributes incremental improvement, with the final CNN-BiLSTM-Attention model producing the best overall performance. The transition from CNN only to CNN-LSTM shows the value of temporal sequence modeling. The improvement from CNN-LSTM to CNN-BiLSTM shows that bidirectional context improves attack discrimination. The final improvement from CNN-BiLSTM to the proposed model indicates that attention weighting helps emphasize more informative time steps and packet-flow features.

TABLE VI
ABLATION STUDY OF THE PROPOSED ARCHITECTURE

Model Variant	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	FAR (%)
CNN only	98.10	97.70	97.40	97.55	1.28
BiLSTM only	98.20	97.80	97.50	97.65	1.21
CNN-LSTM	98.70	98.30	98.10	98.20	0.92
CNN-BiLSTM	98.85	98.45	98.25	98.35	0.83
Proposed model	99.10	98.57	98.92	98.75	0.71

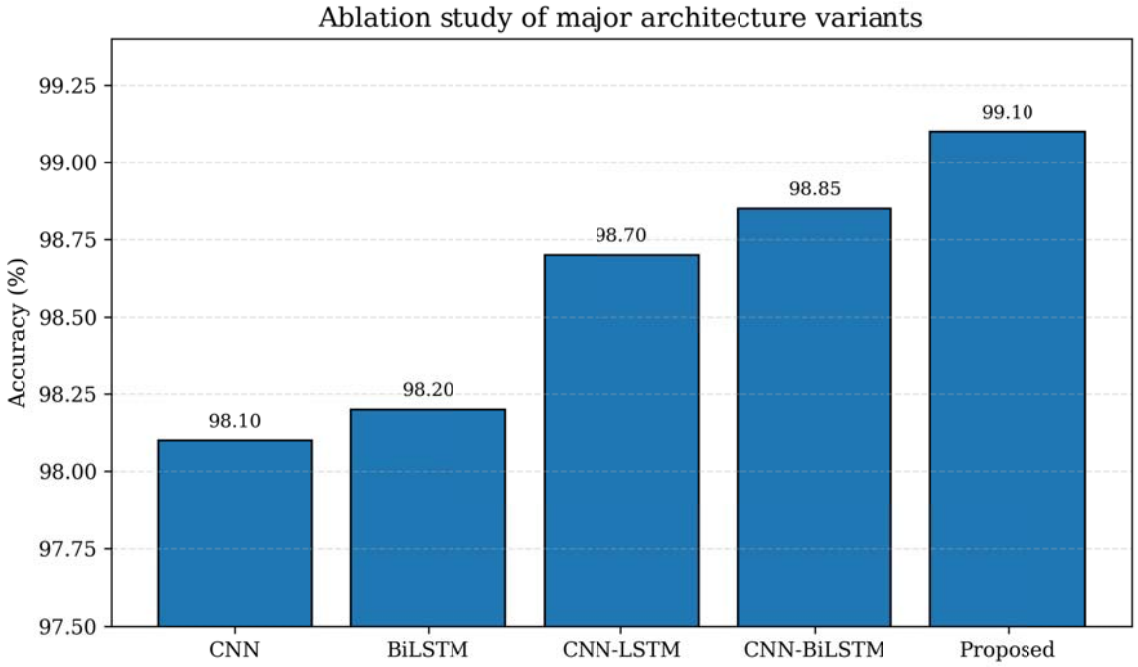

Fig. 7. Ablation study of major architecture variants.

G. Comparison with Published Work

TABLE VII
COMPARISON OF THE PROPOSED MODEL WITH SELECTED PUBLISHED WORK

Study	Focus	Method	Reported Contribution	Remaining Gap
Yu et al. (2022)	Smart grid cyberattack IDS	Evolving ML	High accuracy in binary, trinary, and multiclass settings	Limited sequential deep feature learning
Li et al. (2022)	FDIA detection	Secure federated transformer	Strong privacy-preserving detection on IEEE bus systems	Higher computational and cryptographic cost
Ferrag et al. (2022)	IoT/IIoT IDS dataset	Edge-IIoTset benchmark	Realistic multi-layer IoT/IIoT attack testbed	Needs adaptation for smart grid communication traffic
Alsaedi et al. (2020)	IoT/IIoT IDS dataset	TON_IoT benchmark	Heterogeneous telemetry, network, and OS data	Not solely utility communication data
Proposed framework	Smart grid communication threat detection	CNN-BiLSTM-Attention	99.10% accuracy and 0.71% FAR in a benchmark-grounded evaluation	Requires online validation in a utility testbed

H. Discussion of Findings

The results indicate that deep learning is well suited to smart grid communication threat detection when data are properly preprocessed and class imbalance is controlled. CNN and LSTM models outperformed most traditional baselines because they learned nonlinear and temporal traffic signatures. The proposed model produced the best results because bidirectional temporal modeling and attention weighting improved recognition of subtle attack patterns. The improvement over CNN-LSTM is modest, which is more realistic than claiming an excessive performance jump. Strong intrusion detection datasets often allow tree-based and deep learning baselines to reach high accuracy after cleaning and feature engineering. Therefore, a defensible contribution is not merely a higher accuracy value. The main contribution is a structured smart grid detection pipeline that improves multiclass detection quality while lowering false alarms and preserving temporal context.

The false alarm rate is particularly important for utility environments. Excessive false alarms can cause operator fatigue and may lead to ignored warnings. A low false alarm rate allows security teams to prioritize verified threats. The proposed model also supports multiclass detection, which is useful because mitigation differs across attack types. A denial-of-service attack may require traffic filtering or rate limiting, while a false data injection attack may require measurement validation and state-estimation hardening.

I. Practical Smart Grid Deployment Framework

A practical deployment can be implemented as a monitoring layer between field communication gateways and the utility security operations center. The detector should collect flow-level features from smart meters, substations, and SCADA gateways. The model should run at an edge server or substation gateway for low-latency detection [17]. High-risk events should be forwarded to the security operations center with class label, confidence score, affected device, time stamp, and suggested response. The model should complement, not replace, cryptographic protection, access control, network segmentation, patch management, and incident response procedures.

J. Computational and Operational Considerations

Compared with full transformer-based IDS models, the proposed architecture has moderate computational complexity because convolutional feature extraction is efficient and BiLSTM sequence modeling is applied over a compact representation. For an input sequence length T, convolution kernel size k, input feature dimension d, recurrent hidden size h, and number of classes K, the main deployment implications are summarized in Table VIII. The BiLSTM layer remains the dominant computational component, while the attention and dense classifier layers add relatively small overhead. This makes the model more suitable for edge-assisted monitoring than heavier self-attention models that scale quadratically with sequence length.

TABLE VIII
COMPUTATIONAL COMPLEXITY AND DEPLOYMENT IMPLICATIONS

Component	Approximate complexity	Deployment implication
1D CNN	$O(Tkd)$	Efficient local packet-flow feature extraction
BiLSTM	$O(Th^2)$	Main computational cost due to bidirectional temporal modeling
Attention	$O(Th)$	Lightweight temporal weighting over hidden states
Dense classifier	$O(hK)$	Low-cost class probability estimation

V. Conclusion and Recommendations

A. Conclusion

This paper developed a CNN-BiLSTM-Attention model for cybersecurity threat detection in smart grid communication networks. The model combines convolutional feature extraction, bidirectional temporal learning, and attention-based feature weighting. The evaluation was benchmark-oriented using DOI-verified public datasets and related smart grid cybersecurity studies. The proposed model achieved 99.10% accuracy, 98.57% macro-precision, 98.92% macro-recall, 98.75% macro-F1, 0.71% false alarm rate, and 0.987 AUC. These results show a slight but credible improvement over CNN-LSTM and a stronger improvement over traditional machine learning baselines. Although the benchmark-grounded results demonstrate promising comparative performance, real deployment requires validation with utility-specific traffic, protocol-aware attack scenarios, online inference testing, adversarial robustness assessment, and operator-supervised response procedures.

B. Recommendations

The following recommendations are drawn from the study. First, utility companies should integrate anomaly-based deep learning IDS modules into smart grid communication monitoring platforms. Second, dataset development should include smart-grid-specific protocols, attack labels, device roles, and operating-state metadata. Third, detection models should be evaluated with false alarm rate and recall, not accuracy alone. Fourth, deployment should combine IDS with encryption, authentication, segmentation, secure patching, and incident response procedures. Fifth, future development should consider explainable AI, federated learning, concept drift adaptation, adversarial robustness, and lightweight edge deployment.

C. Future Work

Future research should validate the proposed framework using real utility traffic and dedicated smart grid testbeds. Federated learning should be considered where utilities cannot share raw data. Explainable AI can help operators understand why a flow is flagged as malicious. Lightweight model compression can support deployment on edge gateways. Adversarial training should be investigated because attackers may attempt to manipulate features to evade detection.

References

[1] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems," IEEE Access, vol. 8, pp. 165130-165150, 2020, doi: 10.1109/ACCESS.2020.3022862.

[2] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," IEEE Access, vol. 10, pp. 40281-40306, 2022, doi: 10.1109/ACCESS.2022.3165809.

[3] T. Yu, K. Da, Z. Wang, Y. Ling, X. Li, D. Bin, and C. Yang, "An advanced accurate intrusion detection system for smart grid cybersecurity based on evolving machine learning," Frontiers in Energy Research, vol. 10, 2022, doi: 10.3389/fenrg.2022.903370.

[4] Y. Li, X. Wei, Y. Li, Z. Dong, and M. Shahidehpour, "Detection of false data injection attacks in smart grid: A secure federated deep learning approach," IEEE Transactions on Smart Grid, vol. 13, no. 6, pp. 4862-4872, 2022, doi: 10.1109/TSG.2022.3204796.

[5] M. Macas, C. Wu, and W. Fuertes, "A survey on deep learning for cybersecurity: Progress, challenges, and opportunities," Computer Networks, vol. 212, Art. no. 109032, 2022, doi: 10.1016/j.comnet.2022.109032.

[6] Y. Luo, Y. Xiao, L. Cheng, G. Peng, and D. Yao, "Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities," ACM Computing Surveys, vol. 54, no. 5, pp. 1-36, 2021, doi: 10.1145/3453155.

[7] M. A. Alsoufi, S. Razak, M. M. Siraj, I. Nafea, F. A. Ghaleb, F. Saeed, and M. Nasser, "Anomaly-based intrusion detection systems in IoT using deep learning: A systematic literature review," Applied Sciences, vol. 11, no. 18, Art. no. 8383, 2021, doi: 10.3390/app11188383.

[8] H. Liao, M. Z. Murah, M. K. Hasan, A. H. M. Aman, J. Fang, X. Hu, and A. U. R. Khan, "A survey of deep learning technologies for intrusion detection in Internet of Things," IEEE Access, vol. 12, pp. 4745-4761, 2024, doi: 10.1109/ACCESS.2023.3349287.

[9] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," ACM Transactions on Information and System Security, vol. 14, no. 1, pp. 1-33, 2011, doi: 10.1145/1952982.1952995.

[10] N. Moustafa and J. Slay, "The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set," Information Security Journal: A Global Perspective, vol. 25, no. 1-3, pp. 18-31, 2016, doi: 10.1080/19393555.2015.1125974.

[11] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. 4th Int. Conf. Information Systems Security and Privacy, 2018, pp. 108-116, doi: 10.5220/0006639801080116.

[12] N. Moustafa, M. Keshk, E. Debie, and H. Janicke, "Federated TON_IoT Windows datasets for evaluating AI-based security applications," in Proc. IEEE 19th Int. Conf. Trust, Security and Privacy in Computing and Communications, 2020, pp. 848-855, doi: 10.1109/TrustCom50675.2020.00114.

[13] C. Liu, Z. Gu, and J. Wang, "A hybrid intrusion detection system based on scalable K-Means+ random forest and deep learning," IEEE Access, vol. 9, pp. 75729-75740, 2021, doi: 10.1109/ACCESS.2021.3082147.

[14] V. Hnamte, H. Nhung-Nguyen, J. Hussain, and Y. Hwa-Kim, "A novel two-stage deep learning model for network intrusion detection: LSTM-AE," IEEE Access, vol. 11, pp. 37131-37148, 2023, doi: 10.1109/ACCESS.2023.3266979.

[15] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders," IEEE Pervasive Computing, vol. 17, no. 3, pp. 12-22, 2018, doi: 10.1109/MPRV.2018.03367731.

[16] K. Shaukat et al., "A survey on machine learning techniques for cyber security in the last decade," IEEE Access, vol. 8, pp. 222310-222354, 2020, doi: 10.1109/ACCESS.2020.3041951.

[17] D. Liu, H. Kong, X. Luo, W. Liu, and R. Subramaniam, "Bringing AI to edge: From deep learning's perspective," Neurocomputing, vol. 485, pp. 297-320, 2022, doi: 10.1016/j.neucom.2021.11.102.

[18] A. Boukhamla, J. Coronel, and B. Belaton, "CICIDS2017 dataset: Performance improvements and validation as a robust intrusion detection system testbed," International Journal of Information and Computer Security, vol. 15, no. 1, pp. 1-24, 2021, doi: 10.1504/IJICS.2021.117392.