Federated Intrusion Detection Using TabTransformer-TCN-BiGRU-Attention: A High-Accuracy Hybrid Deep Learning Approach

Noureddine Allassak
Computer Science Department, Faculty of Sciences, IPSS Laboratory, Mohammed V University in Rabat, Rabat, Morocco
noureddine.allassak@um5r.ac.ma (corresponding author)

Salima Trichni
Computer Science Department, Faculty of Sciences, IPSS Laboratory, Mohammed V University in Rabat, Rabat, Morocco | Department of Transversal Modules, Faculty of Law, Economics and Social Sciences, Mohammed V University in Rabat, Rabat, Morocco
s.trichni@um5r.ac.ma

Fouzia Omary
Computer Science Department, Faculty of Sciences, IPSS Laboratory, Mohammed V University in Rabat, Rabat, Morocco
f.omary@um5r.ac.ma

Received: 21 July 2025 | Revised: 10 September 2025 and 28 September 2025 | Accepted: 5 October 2025

 DOI: https://doi.org/10.48084/etasr.13566

ABSTRACT

In the context of Industry 4.0, safeguarding Industrial IoT (IIoT) networks against increasingly sophisticated cyber threats remains a critical challenge, as traditional Intrusion Detection Systems (IDSs) often struggle with scalability, adaptability, and data privacy concerns. This study addresses these limitations by introducing a novel hybrid deep learning architecture for anomaly-based intrusion detection in IIoT environments. The proposed model combines TabTransformer for contextual feature extraction, Temporal Convolutional Networks (TCN) and Bi-directional GRU (BiGRU) for temporal sequence modeling, and an attention mechanism to enhance focus on subtle attack patterns. Using the IoTID20 dataset, the model was first evaluated in centralized training, where it outperformed baseline models (LSTM, CNN, CNN-BiGRU, BiGRU) with an F1-score of 99.8%, accuracy of 99.6%, and an AUC of 0.999. To ensure privacy-preserving and communication-efficient deployment, the model was further implemented in a Federated Learning (FL) setting using Flower, enabling collaborative training across distributed clients without sharing raw data, and significantly reducing bandwidth consumption by exchanging only model parameters. Overall, the proposed approach contributes a scalable, accurate, and privacy-aware intrusion detection framework, positioning hybrid transformer-temporal architectures as promising solutions for secure and intelligent IIoT systems.

Keywords-federated learning; deep learning; TabTransformer; CNN-BiGRU; TCN; intrusion detection

I. INTRODUCTION

The rapid proliferation of Industry 4.0 technologies has significantly increased the reliance on Industrial Internet of Things (IIoT) networks [1], enabling enhanced automation, predictive maintenance, and process optimization [2]. However, this hyperconnectivity also introduces serious cybersecurity vulnerabilities, exposing critical infrastructure to sophisticated cyber-attacks [3]. Protecting IIoT systems has therefore become a priority, with anomaly-based Intrusion Detection Systems (IDSs) emerging as essential components of industrial cybersecurity strategies [4]. Traditional IDS solutions are mainly based on centralized architectures, which require the aggregation of data from multiple sources for training. This approach poses two fundamental challenges in industrial environments: (i) risks of data leakage and violation of privacy

regulations, and (ii) increased communication overhead and single-point-of-failure vulnerabilities. Federated Learning (FL) offers a promising alternative by enabling decentralized training across distributed IIoT clients, where only model parameters are exchanged with a central server, thus preserving data confidentiality [5]. To address the limitations of existing approaches, this study introduces a novel hybrid deep learning model specifically designed for anomaly-based intrusion detection in IIoT networks. The proposed architecture integrates:

- TabTransformer [6] layers to encode feature interdependencies in tabular traffic data;
- Temporal Convolutional Networks (TCN) [7] to model temporal patterns across varying time scales.
- Bidirectional Gated Recurrent Units (BiGRU) to capture forward and backward sequential dependencies.
- A global attention pooling mechanism to enhance feature aggregation and discrimination.

The novelty and contributions of this work are summarized as follows:

- Novel hybrid architecture: A hybrid IDS model integrates TabTransformer, TCN, BiGRU, and an attention mechanism, enabling the joint modeling of contextual and temporal dependencies in IIoT traffic.
- Benchmarking against baselines: The model is systematically evaluated against widely used deep learning baselines (LSTM, CNN, CNN-BiGRU, BiGRU) in a centralized learning setting using the IoTID20 dataset [8], demonstrating clear performance improvements.
- Privacy-aware deployment: To address real-world constraints, the model is deployed within an FL framework using Flower [9], allowing collaborative privacy-preserving training across distributed IIoT clients without sharing raw data.
- Practical validation: Experimental results show that the proposed model achieves superior performance in centralized mode (F1-score 99.8%, AUC 0.999, and Accuracy 99.6%) while maintaining competitive results in federated settings, confirming its scalability and industrial applicability.
- End-to-end feature learning: Unlike most existing IDS works that depend on explicit feature selection techniques, the proposed model autonomously extracts informative patterns from raw preprocessed input through its transformer and attention-based architecture, reducing reliance on manual feature engineering and enhancing generalization.

II. RELATED WORK

Several recent studies have explored the application of FL for intrusion detection in IoT/IIoT contexts. In [10], a federated framework employed ensemble-based feature selection combined with CNN and GRU models for DoS attack detection using the IoTID20 dataset, highlighting the performance benefits of feature selection in FL scenarios. A similar direction was pursued in [11], introducing a lightweight FL-enabled IDS for Wireless Sensor Networks that employed optimized CNN+LSTM models to balance detection performance and resource consumption. In [12], a CNN-based FL system was designed for DDoS attack detection, achieving performance comparable to centralized learning, while addressing non-IID data challenges through model personalization. In [13], an FL-IDS was presented for vehicular networks, utilizing CNN and logistic regression classifiers deployed on edge devices to ensure low-latency, privacy-preserving detection. Beyond model architecture innovations, the work in [14] focused on computational efficiency, presenting a lightweight federated IDS using pruning techniques and deep CNN backbones (ResNet, VGG) to reduce model size without compromising detection accuracy.

In [15], a lightweight hybrid deep learning-based IDS was proposed, tailored for IoT environments. The model integrates CNN with LSTM layers to balance detection accuracy and computational efficiency. Evaluated on the UNSW-NB15 dataset, the hybrid CNN+LSTM architecture achieved an accuracy of 98.78% and an F1-score of 97.99%, while being deployable on resource-constrained devices such as the Raspberry Pi3. This work highlights the effectiveness of combining CNN and LSTM for real-time intrusion detection in low-power IoT infrastructures. In [16], SDN-assisted FL environments were explored, introducing a contrastive learning-based IDS that combines CNN-BiGRU and LSTM-SVM classifiers for robust DDoS attack mitigation. Despite the demonstrated efficacy of these approaches, most of them rely on conventional architectures, such as CNNs, LSTMs, and GRUs, which may not fully capture complex temporal and contextual dependencies within IIoT traffic data. Moreover, these models often depend on manual feature selection processes to optimize performance, potentially limiting their adaptability. Table I presents a comparative summary of related FL-based intrusion detection approaches, highlighting their datasets, methodologies, models, learning environments, and reported performances.

III. METHODOLOGY

This study aimed to develop an advanced intrusion detection framework tailored for IIoT environments. The proposed approach leverages a novel hybrid deep learning architecture that combines TabTransformer, TCN, BiGRU, and an Attention mechanism to enhance anomaly detection performance [17].

A. System Architecture Overview

Figure 1 shows the overall architecture of the proposed federated IDS. Multiple IIoT clients locally train a shared detection model without exchanging raw data. Model updates are sent to a central server, where Federated Averaging (FedAvg) [18] aggregates the parameters into a global model, which is then redistributed for continuous training.

TABLE I. COMPARATIVE ANALYSIS OF RELATED FL-BASED INTRUSION DETECTION METHODS

Study	Domain / Dataset	Methodology	Models Used	Feature selection	FL environment	Performance
[10]	IoTID20 / DoS detection	FL + Ensemble feature selection + DL	RNN, LSTM, GRU, CNN	Yes (Ensemble + RFE)	FL + Centralized (comparison)	FL Accuracy: 99.73% (CNN + ANOVA)
[11]	TON-IoT / DDoS detection (WSN)	FL with lightweight models	CNN, LSTM (hybrid)	No	FL	F1-score ≈ 99%
[12]	DDoS dataset / DDoS detection	FL with CNN	Simple CNN	No	FL	Accuracy ≈ 99%
[13]	NSL-KDD / Car-Hacking	FL using edge devices	Logistic regression, CNN	No	FL	Accuracy: 94% – 99%
[14]	UNSW-NB15, CIC-IDS2017	FL + Model pruning (Lightweight-FL)	ResNet, VGG, CNN	No (Pruning applied)	FL	Detection rate ≈ 99%
[15]	UNSW-NB15 / IoT intrusion detection	Lightweight hybrid DL	CNN + LSTM (hybrid)	No	Centralized (Edge device)	Accuracy ≈ 98.78%, F1-score ≈ 97.99%
[16]	SDN Dataset / DDoS detection	Contrastive learning + SDN-assisted FL	CNN-BiGRU, LSTM-SVM	No	SDN-assisted FL	CNN-BiGRU Accuracy: 99.36%
This study	IoTID20 / Intrusion detection	Hybrid DL + FL	TabTransformer + TCN + BiGRU + Attention	No (only normalization)	FL + Centralized (comparison)	Accuracy.> 99.6%, F1-score FL ≈ 99.8%

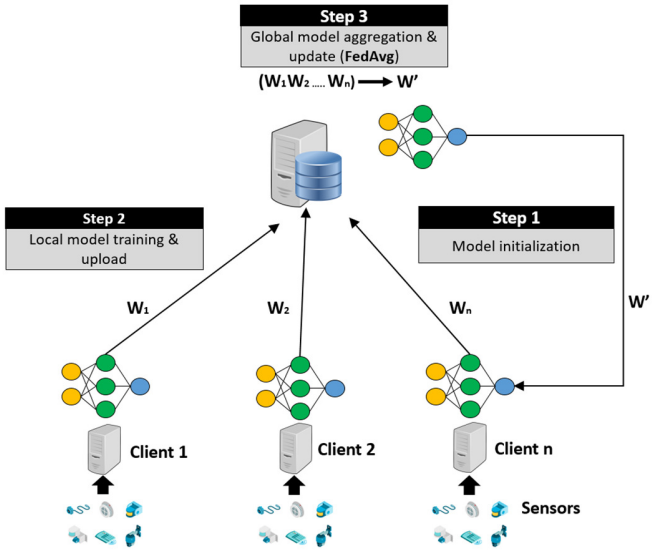

Fig. 1. FL architecture for privacy-preserving IIoT intrusion detection.

Figure 2 details the workflow of the proposed hybrid deep learning model. After preprocessing, the model combines TabTransformer layers, TCN, BiGRU, and attention pooling to extract rich temporal and contextual features from IIoT traffic. This pipeline supports both centralized evaluation and FL deployment while ensuring data privacy.

B. Data Description

This study uses the IoTID20 dataset, which contains 461,696 labeled network traffic records (Normal and Attack) [8]. Each record is characterized by flow-based, packet-based, and time-based features. The dataset exhibits heterogeneous distributions with extreme values across several features (e.g., packet lengths and idle times), which reflects the diversity of IoT traffic. This variability, combined with class imbalance, makes IoTID20 a challenging benchmark for intrusion detection. Table II summarizes the main attributes and their statistical ranges.

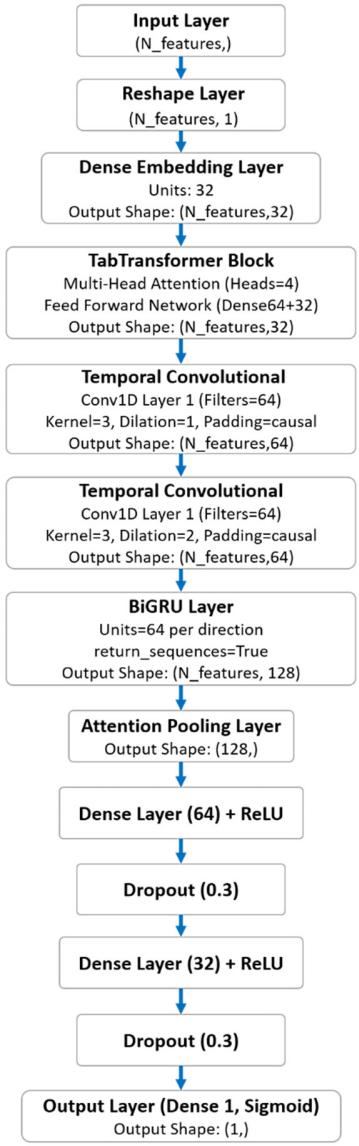

Fig. 2. Proposed hybrid IDS pipeline integrating TabTransformer, TCN, BiGRU, and Attention mechanisms.

TABLE II. DESCRIPTION AND STATISTICAL SUMMARY OF THE MAIN ATTRIBUTES OF THE DATASET

Feature	Min	Max	Mean	SD	Description
Flow_Duration	0	99.98	811.77	4056.13	Duration of the flow (ms)
Tot_Fwd_Pkts	0	186	1.94	4.96	Total forward packets
Tot_Bwd_Pkts	1	560	1.49	1.36	Total backward packets
TotLen_Fwd_Pkts	0	109.84	465.07	1167.70	Bytes in forward packets
TotLen_Bwd_Pkts	0	773.28	792.21	1898.31	Bytes in backward packets
Active_Mean	0	9044.63	5,10	79.19	Mean active time per flow
Idle_Mean	0	99.973	642.45	2444.49	Mean idle time per flow

C. Data Preprocessing

Before model training, the dataset underwent a structured preprocessing phase to enhance learning efficiency and model generalization. Irrelevant or non-informative columns, such as flow identifiers and IP address fields, were removed from the dataset. Specifically, the columns 'Flow_ID', 'Src_IP', 'Dst_IP', 'Timestamp', 'Cat', and 'Sub_Cat' were discarded, as they provide no meaningful contribution to anomaly detection tasks and could introduce unnecessary complexity.

Following column elimination, any missing or infinite values were handled to ensure data integrity. Subsequently, all numerical features were standardized using z-score normalization via the StandardScaler function from scikit-learn, ensuring that each feature contributes equally to the learning process [19]. Feature selection techniques were deliberately excluded, as the hybrid model's architecture is designed to learn relevant feature representations autonomously, relying on its transformer and attention mechanisms to identify informative patterns directly from the preprocessed input.

D. Hybrid Model Architecture

The proposed model leverages a streamlined hybrid deep learning architecture to analyze IIoT traffic data efficiently. As illustrated in Figure 2, the pipeline begins with reshaping the tabular inputs into a sequential format, enabling temporal analysis. TabTransformer layers are applied first to capture dependencies between features using multi-head attention, enhancing the raw input representation without manual feature selection. These enriched embeddings are then processed through two stacked TCN blocks, which learn localized and long-range temporal patterns through causal convolutions and dilation. To capture sequential dependencies more comprehensively, the TCN output is passed through a BiGRU layer, which processes data in both temporal directions. An attention pooling layer follows, weighting the most informative temporal features for anomaly detection. Finally, fully connected dense layers and dropout are used for feature consolidation and regularization, ending with a sigmoid layer for binary classification. This hybrid architecture combining TabTransformer, TCN, BiGRU, and attention mechanisms ensures robust feature learning directly from raw IIoT traffic, supporting both centralized and federated training scenarios. To clarify the contribution of each architectural block, Table III summarizes the role of the core components within the proposed hybrid model.

TABLE III. CORE COMPONENTS OF THE PROPOSED HYBRID DEEP LEARNING MODEL

Module	Role
TabTransformer	Feature embedding and interaction modeling.
Temporal Convolutional Networks (TCN)	Temporal pattern extraction.
Bi-directional GRU (BiGRU)	Sequential dependency learning (both directions).
Attention pooling	Focus on salient temporal features.
Dense layers	Final classification decision.

E. Centralized Training

Initially, the model was trained in a centralized setup to benchmark its performance against standard deep learning architectures, including LSTM, CNN, CNN-BiGRU, and BiGRU-only models. Performance evaluation employed Accuracy (1), Precision (2), Recall (3), F1-score (4), and AUC-ROC metrics. To ensure fair benchmarking, the proposed model's architecture was fine-tuned using Grid Search [20]. Table IV summarizes the range of hyperparameters explored, while Table V lists the final configuration adopted for centralized training.

TABLE IV. GRID SEARCH PARAMETER RANGES

Block / Layer	Hyperparameter	Explored values
TabTransformer Block	embed_dim	16, 32, 64
	num_heads	2, 4, 8
	ff_dim	32, 64, 128
TCN Block	filters	32, 64, 128
	kernel_size	2, 3, 5
	dilation_rate	1, 2, 4
BiGRU Layer	units	32, 64, 128
Dense Layers	units (Dense Layers)	32, 64, 128
	dropout_rate	0.2, 0.3, 0.5
Optimization	batch_size	32, 64, 128
	learning_rate (Adam)	1e-3, 5e-4, 1e-4

TABLE V. SELECTED OPTIMAL HYPERPARAMETERS

Block / Layer	Hyperparameters	Optimal value
TabTransformer Block	embed_dim	32
	num_heads	4
	ff_dim	64
TCN Block (x2)	filters	64
	kernel_size	3
	dilation_rate	1 (1st bloc), 2 (2nd bloc)
BiGRU Layer	units	64
Dense Layer 1	units	64
Dropout Layer 1	dropout_rate	0,3
Dense Layer 2	units	32
Dropout Layer 2	dropout_rate	0,3
Output Layer	Activation	Sigmoid
Optimization	learning_rate	0,001
	batch_size	128

For consistency, all comparative models were trained under the same optimization settings, namely the Adam optimizer with a learning rate of 0.001, binary cross-entropy loss, and

accuracy as the evaluation metric. This uniform configuration was applied intentionally to ensure fair comparisons between models rather than favoring a specific architecture through selective hyperparameter tuning. The detailed configurations of each model are reported in Table VI. All models were implemented using the Keras 3.10.0 API with TensorFlow 2.19.0 as the backend, ensuring consistent training conditions. The dataset was randomly divided into training (80%) and testing (20%) subsets, preserving the original distribution of normal and anomalous samples to avoid class imbalance issues. This guarantees that performance comparisons across models are based on the same experimental protocol.

TABLE VI. BASELINE MODEL ARCHITECTURES USED FOR COMPARATIVE EVALUATION

Model	Architecture Description	Optimization
LSTM	Reshape (features,1) → LSTM (64 units) → Dense (32, ReLU) → Dropout (0.3) → Dense (1, Sigmoid)	**Optimizer:** Adam (Learning rate=10^{-3}). **Loss Function:** Binary cross-rntropy.
CNN	Reshape (features,1) → Conv1D (64 filters, kernel size=3, ReLU) → MaxPooling1D (pool=2) → Flatten → Dense (64, ReLU) → Dropout (0.3) → Dense (1, Sigmoid)	
BiGRU	Reshape (features,1) → Bidirectional GRU (64 units) → Dense (32, ReLU) → Dropout (0.3) → Dense (1, Sigmoid)	
CNN-BiGRU	Reshape (features,1) → Conv1D (32 filters, kernel size=3, ReLU) → Bidirectional GRU (64 units, return sequences=True) → Flatten → Dense (32, ReLU) → Dropout (0.3) → Dense (1, Sigmoid)	

F. Federated Learning (FL) Deployment

To address data privacy challenges in IIoT environments, the hybrid model was adapted to an FL framework using the Flower library. In this decentralized setup, three simulated clients train local model instances on distinct data partitions. Only model weights are shared with a central server for aggregation, preserving data confidentiality. Evaluation metrics on each client validate the consistency of detection performance across distributed nodes.

G. Evaluation Metrics

To rigorously evaluate the classification performance of the proposed intrusion detection model, five widely adopted metrics were utilized: Accuracy, Precision, Recall, F1-score, and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC). These metrics collectively provide a comprehensive understanding of the model's effectiveness in detecting anomalous activities within IIoT traffic. Let TP (True Positives), TN (True Negatives), FP (False Positives), and FN (False Negatives) represent the confusion matrix components [21]. The evaluation metrics are formally defined as follows:

- Accuracy measures the proportion of correct predictions among all predictions:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

- Precision quantifies the proportion of correctly identified positive instances among all instances predicted as positive:

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

- Recall (also called Sensitivity) evaluates the proportion of actual positives correctly identified by the model:

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

- F1-score represents the harmonic mean of Precision and Recall, providing a balanced assessment:

$$F1-score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (4)$$

- AUC-ROC measures the model's ability to discriminate between positive and negative classes across different thresholds. It is calculated as the area under the ROC curve, where higher values indicate stronger discriminatory performance. These metrics were chosen to ensure a robust evaluation, particularly under class-imbalanced scenarios typical in anomaly detection tasks.

IV. RESULTS AND DISCUSSION

The performance of the proposed hybrid deep learning model was evaluated in both centralized and FL frameworks to assess its scalability and robustness. In the FL scenario, three IIoT clients participated across three communication rounds, with model updates aggregated using FedAvg. Tables VII and VIII report detailed evaluation metrics, including Accuracy, Precision, Recall, F1-score, and AUC-ROC for all compared models under both centralized and federated setups. Additionally, Figures 3 and 4 graphically highlight the comparative Accuracy and F1-score between centralized training and the averaged results from federated clients.

TABLE VII. CENTRALIZED TRAINING: PERFORMANCE COMPARISON OF EVALUATED MODELS

Model	Accuracy	Precision	Recall	F1-score	AUC-ROC
Proposed model	0.9969	0.9973	0.9993	0.9983	0.9991
BiGRU	0.9969	0.9971	0.9996	0.9984	0.9990
CNN-BiGRU	0.9968	0.9966	1	0.9983	0.9991
CNN	0.9949	0.9949	0.9996	0.9972	0.9968
LSTM	0.9837	0.9862	0.9962	0.9911	0.9891

TABLE VIII. FL: PERFORMANCE OF MODELS ACROSS INDIVIDUAL CLIENTS

Model	Clt	Accur.	Precision	Recall	F1-score	AUC-ROC
Proposed model	1	0.9967	0.9966	0.9999	0.9982	0.9987
	2	0.9969	0.9969	0.9997	0.9983	0.999
	3	0.9966	0.9963	0.9999	0.9981	0.9996
BIGRU	1	0.9930	0.9930	0.9996	0.9963	0.9985
	2	0.9919	0.9918	0.9995	0.9956	0.9988
	3	0.9919	0.9919	0.9992	0.9955	0.9991
CNN-BIGRU	1	0.9956	0.9957	0.9996	0.9977	0.9986
	2	0.9953	0.9955	0.9994	0.9974	0.9992
	3	0.9944	0.9942	0.9996	0.9969	0.9995
CNN	1	0.9955	0.9954	0.9998	0.9976	0.9951
	2	0.9949	0.9949	0.9996	0.9972	0.9971
	3	0.9945	0.994	1	0.997	0.9974
LSTM	1	0.9368	0.9648	0.9674	0.9661	0.8535
	2	0.9367	0.9575	0.9742	0.9658	0.8666
	3	0.9276	0.9496	0.9713	0.9603	0.8689

In centralized training, the proposed TabTransformer+TCN+BiGRU+Attention model achieved superior performance, with an F1-score exceeding 99.8% and an AUC-ROC approaching 0.999, slightly outperforming BiGRU and CNN-BiGRU architectures. Traditional models such as LSTM and CNN exhibited comparatively lower scores, confirming the importance of hybrid feature extraction mechanisms. The confusion matrix (Figure 5) confirms the model's precision and reliability by illustrating a negligible number of false positives and false negatives.

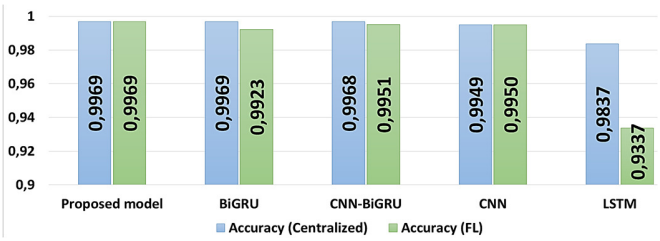

Fig. 3. Centralized vs Federated Learning: Accuracy comparison across models.

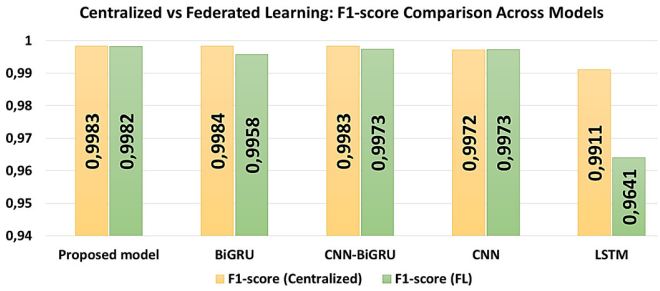

Fig. 4. Centralized vs Federated Learning: F1-score comparison across models.

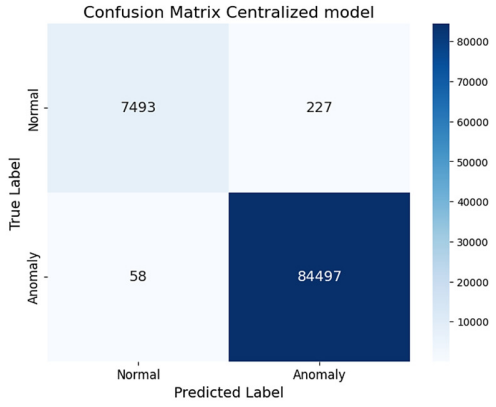

Fig. 5. Confusion matrix of the proposed model in centralized mode.

In the federated setting, using FedAvg, the proposed hybrid model maintained comparable detection capabilities across all three clients, with only marginal performance drops compared to centralized training. Metrics such as F1-score and AUC-ROC consistently exceeded 0.998 across all clients, demonstrating the model's ability to generalize despite decentralized, privacy-preserving training. Confusion matrices for each client (Figures 6, 7, and 8) corroborate these findings, showing balanced and accurate predictions.

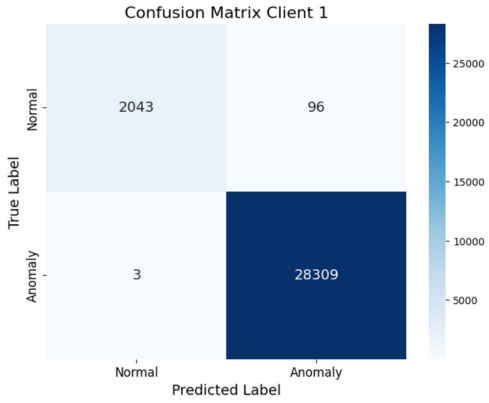

Fig. 6. Confusion matrix of Client 1 in FL.

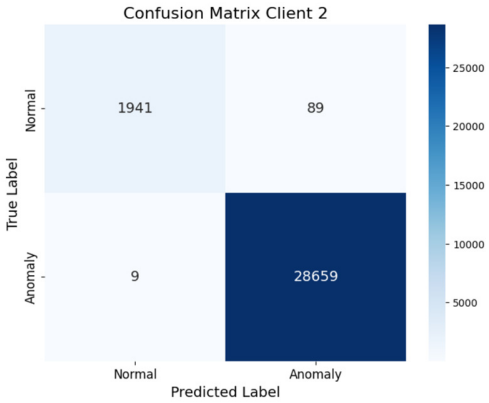

Fig. 7. Confusion Matrix of Client 2 in FL.

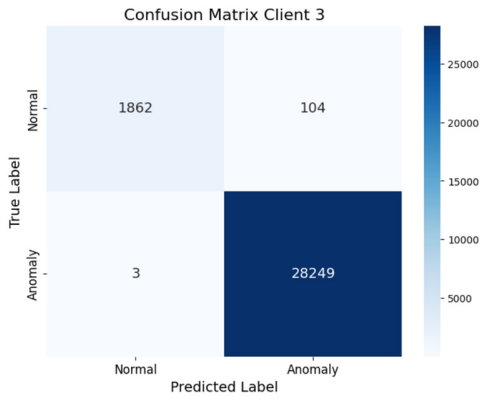

Fig. 8. Confusion Matrix of Client 2 in FL.

Comparatively, models such as BiGRU and CNN-BiGRU showed minor performance degradations in the FL context, particularly on clients with slightly smaller data partitions. In contrast, LSTM exhibited the most significant drop in federated mode, confirming its limitations when handling non-IID data in distributed environments.

Overall, this study reveals that combining TabTransformer with temporal sequence learners (TCN and BiGRU) and an attention mechanism provides a robust approach for intrusion detection in IIoT environments. The proposed model achieves better performance compared to classical deep learning models, proving its capacity to extract richer feature representations. In addition, its successful deployment in FL demonstrates practical benefits, namely scalability, privacy preservation, and adaptability to distributed industrial settings. These findings confirm the relevance of the model to more secure, privacy-aware, and communication-efficient IDSs for Industry 4.0.

Beyond empirical performance, these findings provide two broader implications for the research community. First, they uncover critical directions by showing that hybrid transformer–temporal architectures are particularly effective in capturing both contextual and sequential dependencies in IIoT traffic, thus offering researchers a concrete pathway for future IDS benchmarking. Second, the successful integration of this architecture within an FL framework advances the theoretical understanding that privacy preservation, scalability, and high accuracy are not mutually exclusive goals in distributed intrusion detection. This is achieved by drastically reducing the bandwidth overhead associated with transferring massive raw datasets. This contributes to the emerging theory that hybrid representation learning, when coupled with communication-efficient federated aggregation, can serve as a foundation for next-generation IDS research in Industry 4.0.

V. CONCLUSION

This study presented a novel hybrid deep learning architecture for anomaly-based intrusion detection in IIoT networks. Unlike prior works that typically relied on either recurrent or convolutional models, this one integrates TabTransformer-based contextual feature extraction, temporal modeling via TCN and BiGRU, and an attention mechanism to enhance the detection of subtle attack patterns. This synergy enables the model to capture both complex feature dependencies and temporal dynamics, addressing limitations observed in conventional IDS models. A distinctive aspect of this approach lies in the deliberate exclusion of traditional feature selection techniques. Instead, the hybrid architecture autonomously learns and prioritizes relevant feature representations through its transformer and attention mechanisms, enhancing generalizability and reducing reliance on manual feature engineering. Extensive experiments demonstrated that the proposed model consistently outperformed state-of-the-art baselines (LSTM, CNN, CNN-BiGRU, BiGRU-only) in centralized training, achieving higher accuracy, F1-score, and AUC-ROC. More importantly, its deployment in an FL setup confirmed scalability and robustness while preserving privacy, as raw data remain local, and reducing communication overhead by exchanging only model updates instead of massive datasets. By bridging advanced feature representation with privacy-aware and communication-efficient deployment, this work contributes a new perspective to intrusion detection research, positioning hybrid transformer-temporal architectures as strong candidates for building secure, intelligent, and practical IDS solutions in Industry 4.0 environments.

REFERENCES

[1] N. T. Ching, M. Ghobakhloo, M. Iranmanesh, P. Maroufkhani, and S. Asadi, "Industry 4.0 applications for sustainable manufacturing: A systematic literature review and a roadmap to sustainable development," *Journal of Cleaner Production*, vol. 334, Feb. 2022, Art. no. 130133, https://doi.org/10.1016/j.jclepro.2021.130133.

[2] M. Achouch *et al.*, "On Predictive Maintenance in Industry 4.0: Overview, Models, and Challenges," *Applied Sciences*, vol. 12, no. 16, Aug. 2022, Art. no. 8081, https://doi.org/10.3390/app12168081.

[3] K. Tsiknas, D. Taketzis, K. Demertzis, and C. Skianis, "Cyber Threats to Industrial IoT: A Survey on Attacks and Countermeasures," *IoT*, vol. 2, no. 1, pp. 163–186, Mar. 2021, https://doi.org/10.3390/iot2010009.

[4] O. H. Abdulganiyu, T. Ait Tchakoucht, and Y. K. Saheed, "A systematic literature review for network intrusion detection system (IDS)," *International Journal of Information Security*, vol. 22, no. 5, pp. 1125–1162, Oct. 2023, https://doi.org/10.1007/s10207-023-00682-2.

[5] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, Mar. 2021, Art. no. 106775, https://doi.org/10.1016/j.knosys.2021.106775.

[6] X. Huang, A. Khetan, M. Cvitkovic, and Z. Karnin, "TabTransformer: Tabular Data Modeling Using Contextual Embeddings." arXiv, 2020, https://doi.org/10.48550/ARXIV.2012.06678.

[7] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal Convolutional Networks for Action Segmentation and Detection." arXiv, 2016, https://doi.org/10.48550/ARXIV.1611.05267.

[8] I. Ullah and Q. H. Mahmoud, "A Scheme for Generating a Dataset for Anomalous Activity Detection in IoT Networks," in *Advances in Artificial Intelligence*, 2020, pp. 508–520, https://doi.org/10.1007/978-3-030-47358-7_52.

[9] D. J. Beutel *et al.*, "Flower: A Friendly Federated Learning Research Framework." arXiv, 2020, https://doi.org/10.48550/ARXIV.2007.14390.

[10] T. Q. Al-Ghadi, S. Manickam, I. D. M. Widia, E. R. N. Wulandari, and S. Karuppayah, "Leveraging federated learning for DoS attack detection in IoT networks based on ensemble feature selection and deep learning models," *Cyber Security and Applications*, vol. 3, Dec. 2025, Art. no. 100098, https://doi.org/10.1016/j.csa.2025.100098.

[11] M. Devi, P. Nandal, and H. Sehrawat, "Federated learning-enabled lightweight intrusion detection system for wireless sensor networks: A cybersecurity approach against DDoS attacks in smart city environments," *Intelligent Systems with Applications*, vol. 27, Sep. 2025, Art. no. 200553, https://doi.org/10.1016/j.iswa.2025.200553.

[12] D. Lv, X. Cheng, J. Zhang, W. Zhang, W. Zhao, and H. Xu, "DDoS Attack Detection Based on CNN and Federated Learning," in *2021 Ninth International Conference on Advanced Cloud and Big Data (CBD)*, Xi'an, China, Mar. 2022, pp. 236–241, https://doi.org/10.1109/CBD54617.2021.00048.

[13] M. H. Bhavsar, Y. B. Bekele, K. Roy, J. C. Kelly, and D. Limbrick, "FL-IDS: Federated Learning-Based Intrusion Detection System Using Edge Devices for Transportation IoT," *IEEE Access*, vol. 12, pp. 52215–52226, 2024, https://doi.org/10.1109/ACCESS.2024.3386631.

[14] A. Bouayad, H. Alami, M. Janati Idrissi, and I. Berrada, "Lightweight Federated Learning for Efficient Network Intrusion Detection," *IEEE Access*, vol. 12, pp. 172027–172045, 2024, https://doi.org/10.1109/ACCESS.2024.3494057.

[15] R. H. Altaie and H. K. Hoomod, "An Intrusion Detection System using a Hybrid Lightweight Deep Learning Algorithm," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 16740–16743, Oct. 2024, https://doi.org/10.48084/etasr.7657.

[16] M. Fan, J. Lan, Y. Zhou, M. Pan, J. Li, and D. Zhang, "DDoS Attack Detection in SDN-Assisted Federated Learning Environment Based on Contrastive Learning," *IEEE Access*, vol. 13, pp. 108798–108814, 2025, https://doi.org/10.1109/ACCESS.2025.3582445.

[17] Z. Niu, G. Zhong, and H. Yu, "A review on the attention mechanism of deep learning," *Neurocomputing*, vol. 452, pp. 48–62, Sep. 2021, https://doi.org/10.1016/j.neucom.2021.03.091.

[18] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni, "Federated Learning with Matched Averaging." arXiv, 2020, https://doi.org/10.48550/ARXIV.2002.06440.

[19] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," 2012, https://doi.org/10.48550/ARXIV.1201.0490.

[20] P. Liashchynskyi and P. Liashchynskyi, "Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS." arXiv, 2019, https://doi.org/10.48550/ARXIV.1912.06059.

[21] K. M. Ting, "Confusion Matrix," in *Encyclopedia of machine learning*, Springer Science & Business Media, 2011.