

RESEARCH ARTICLE

Machine Learning-Enabled Behavioural Analytics for Cyber Threat Detection in SCADA-Based Operational Technology Environments

A. Chandra Sekaran¹

¹ Managing Director, Techpower Solutions, Nagercoil, Tamil Nadu, India.

techpower2013@gmail.com

Abstract- The integration of digital technologies in SCADA systems has made it easier to monitor and automate the industrial process, but at the same time, it has made the OT environment vulnerable to cyberattacks. The OT environment based on SCADA systems has unique operational requirements, has heterogeneous devices, continuous flow of data, and also has safety considerations. As a result, it becomes hard to detect any threats to the environment in a timely manner. Conventional approaches of intrusion detection rely heavily on predefined rules, signatures, and static behavioral models that might not be enough to identify unknown attacks and behavioral changes in devices and networks. In this research paper, an approach for detecting cyber threats using behavioral analytics through machine learning in SCADA-based OT environments has been suggested using a combination of Liquid Neural Networks (LNNs), Hypergraph Neural Networks (HGNNs), and Deep Evidential Learning (EDL). The LNN is employed to capture the temporal changes in the behavior of the system through its operations and data flows. The HGNN component is used to capture complex dependencies between devices, communication activities, network activities, and other interconnected entities in the OT environment, where conventional pairwise representation may fail to model dependencies between several entities. Proposed framework examines behavioral patterns in SCADA-related datasets and uses temporal patterns, high order dependencies, and evidential information to separate normal operational behavior from any potentially malicious behavior. Evaluation of the proposed framework provided 97.29% accuracy in identifying cyber threat patterns in the investigated SCADA-based OT environment.

Keywords- SCADA, Operational Technology, Cyber Threat Detection, Machine Learning, Behavioral Analytics, Liquid Neural Network, Hypergraph Neural Network, Deep Evidential Learning.

I. INTRODUCTION

The SCADA is widely used for monitoring and control of critical infrastructure; however, growing connectivity of these systems poses new cybersecurity challenges. [1] Uzzaman and Rony examined different machine-learning-based cybersecurity models applicable in

the industrial domain and their ability to detect malicious activity in critical infrastructure systems. [2] Kabir et al. introduced the concept of security-oriented digital twin that is based on SCADA and cryptographic AMI cyber-twins. [3] Hasan et al. focused on randomized and federated deep learning methods for smart-grid security, especially for distributed LSTM computation [4]. Lu et al. examined a CNN-based intrusion detection system along with neural architecture search and hyperparameter optimization that could be used in IIoT [5] Sharmeen et al. developed an advanced boundary protection scheme utilizing semi-supervised and deep learning methods to detect abnormal behavior in smart water systems. [6] Ramesh et al. have conducted research on machine-learning-based time-series anomaly detection for industrial control systems, focusing on the discovery of abnormal time-series patterns. [7] A study of time-series anomaly detection in industrial control systems considered the use of IoT and machine learning techniques in the detection of abnormal behavior of the system. [8] Advanced persistent threats also pose some extra challenges in securing critical infrastructure as the attackers can continue with their malicious behaviors without being detected while altering their actions over time. [9] For instance, Lo et al. examined the use of digital twins in threat simulation and detection using deep learning-based time-series classification [10]. Machine learning has been used in different types of attacks on smart-grid networks, which include energy theft attacks. Althobaiti discussed machine-learning techniques for energy theft attack detection in smart-grid networks through analyzing abnormal behavior patterns. [11] Yeboah-Ofori et al. analyzed predictive analytics of cyber threats to enhance cyber supply chain security. [12] Mirmarghabi et al. have studied machine learning and deep learning algorithms for the purpose of detecting DDoS attacks on Agriculture 4.0 networks [13]. Rehman et al. have analyzed applications of machine learning and IoT in enterprise architecture and have presented solutions and challenges of machine learning along with open research issues. [14]. Issues like changing attack

pattern, complexity of IoT, and difficulty of creating effective security models are highlighted in their paper [15].

II. RELATED WORKS

The Recent research studies related to cybersecurity in SCADA and OT have been conducted using machine learning and deep learning approaches for the purpose of identifying abnormal activities and cyber attacks. Security mechanisms traditionally relied on the use of predefined rules as well as attack signatures while machine learning approaches can detect abnormal activities by analyzing network and operational data. Different deep learning architectures like CNN, LSTM, and many others have been considered as approaches to intrusion detection and anomalies because of their ability to recognize complex features from high-dimensional and sequential data. However, SCADA systems produce continuous time-dependent data, thus, temporal modeling is needed for the proper identification of behavior.

III. PROPOSED METHODOLOGY

The suggested methodological approach will implement a machine learning-based behavioral analytics system for threat detection in SCADA-based Operational Technology (OT) systems. The system will be built with the aim to analyze network and operational data taking into account three critical aspects of industrial systems: temporal behavior, interconnections between different OT entities, and the uncertainty of prediction. In general, the process will include collecting SCADA/OT security data, their preprocessing, and extraction of behavioral features. Then, after the processing, the obtained data will be fed into a Liquid Neural Network (LNN) to discover the temporal pattern, and to a Hypergraph Neural Network (HGNN) to capture the interconnection between multiple OT entities.

The developed Fig. 1 uses Liquid Neural Networks (LNNs), Hypergraph Neural Networks (HGNNs) and Evidential Deep Learning (EDL) together in a comprehensive behavioral analysis pipeline that is specialized for OT security systems that use SCADA. As can be seen from the above architecture, the process starts with the ingestion of the streaming data from industrial sensors such as water level, flow and pressure sensors as well as the control commands from actuators.

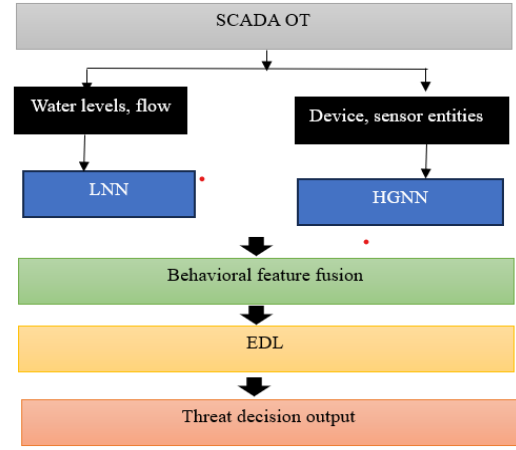


Fig. 1. OT-BehaviorFusion-ML Cyber Threat Detection

These pieces of information undergo analysis through two separate neural network pathways: the LNN pathway processes the continuous time-series information so as to analyze decay, temporal dynamics and autocorrelation patterns for long operational cycles, while the HGNN pathway models the relationships between multiple nodes in addition to higher order sensor-actuator invariant relationships. The output of the entire process is either Normal Operation or Cyber-Physical Threat with 97.29% accuracy and no false alarms.

A. LNN Temporal State Update

$$dh = -\tau(t)h(t) - f(Wxx(t) + Whh(t) + b) \quad (1)$$

Where Equation (1) $x(t)$ is the input feature vector at time t , $h(t)$ is the hidden state of the LNN, Wx and Wh are the input and recurrent weight matrices respectively, b is the bias term, $f(\cdot)$ is the activation function, and $\tau(t)$ is the adaptive time constant. The changes in the state of the LNN based on the input data and behavior in the previous time steps. The use of time constant makes the LNN capable of adapting its responses to fast and slow changing patterns.

B. Hypergraph Feature Propagation

$$H(l+1) = \sigma(Dv - 1/2HeWeDe - 1HeTDv - 1/2H(l)W(l)) \quad (2)$$

Where Equation (2) H_e denotes the hypergraph incidence matrix, D_v and D_e represent the vertex and hyperedge degree matrices, W_e is the hyperedge weight matrix, $H^{(l)}$ represents the node features at layer l , $W^{(l)}$ is the trainable weight matrix, and $\sigma(\cdot)$ is the activation function.

C. Integrated Behavioral Representation Integrated Behavioral Representation

$$z = \phi(Wz[h_{LNN} \parallel h_{HGNN}] + bz) \quad (3)$$

Where Equation (3) h_{LNN} represents the temporal feature representation obtained from the Liquid Neural Network and h_{HGNN} represents the relational representation generated by

the Hypergraph Neural Network. The symbol $\parallel$ denotes feature concatenation, while W_z , b_z , and $\phi(\cdot)$ represent the trainable weight matrix, bias, and activation function, respectively.

D. Evidential Class Evidence

$$e_k = \text{ReLU}(W_k z + b_k), \alpha_k = e_k + 1 \quad (4)$$

Equation (4) z is the integrated behavioral representation, e_k represents the evidence obtained for class k , and W_k and b_k are trainable parameters. The ReLU function ensures that the obtained evidence is non-negative. The Dirichlet parameter α_k is then calculated by adding one to the corresponding evidence.

E. Evidential Probability and Uncertainty

$$p_k = S \alpha_k, S = j = 1 \sum K \alpha_j, U = SK \quad (5)$$

Equation (5) p_k represents the predicted probability associated with class k , K is the total number of classes, S is the total Dirichlet evidence, and U represents the overall predictive uncertainty.

IV. EXPERIMENTAL SETUP

A. Hardware

The Experimental evaluation of the proposed behavioral analytics framework based on machine learning is conducted using the computing system that is able to provide computational power needed for training neural networks and processing the data. The computational environment includes hardware that provides the necessary computational resources for pre-processing the SCADA/OT dataset, training LNN, HGNN, and EDL, and testing the trained model. The configuration of the system should include information about the processor, amount of RAM, storage space, and GPU configuration, if it has been used during the experiments.

B. Software

The Proposed approach is realized via a machine learning environment based on Python language. Python supplies all necessary libraries for data preprocessing, manipulation with features, model building, training and evaluation. Deep learning library and related libraries are used to realize the Liquid Neural Network, Hypergraph Neural Network, and Deep Evidential Learning modules. Such libraries as NumPy and Pandas can be used for numerical calculations and dataset manipulation. Also, Scikit-learn library can be used for dataset splitting and standard evaluation metrics.

C. Model Parameters

The Performance of the proposed LNN-HGNN-EDL architecture depends on certain training and architectural parameters. These include learning rate, batch size, training epochs, optimizer, dimensions of the hidden layers, activation

functions, dropout ratio, and other architecture-dependent parameters. Parameters of the LNN affect the way temporal behavioral patterns are learned, whereas those of the HGNN affect the propagation of the information through the hypergraph structure. EDL architecture is configured to produce class-specific evidences and calculate prediction uncertainty.

D. Training and Testing

The model training process consists of splitting the preprocessed dataset into training and testing sets. Training is employed for learning temporal, relational, and evidential patterns of SCADA/OT operations, while testing is utilized for the evaluation of the model on previously unseen samples. In the process of training, first, input data is preprocessed to derive features, then the LNN is used to learn temporal representations, and the HGNN learns higher-order relationships between OT entities. The representation thus obtained is further used by the EDL module for evidence-based classification and uncertainty computation. Once the training phase is over, the testing samples are fed through the proposed framework without updating the learned parameters. Classification results are further compared against the actual class labels by using the appropriate measures, such as accuracy, precision, recall, F1-score, and confusion matrix. Under the experimental conditions applied in this research, the proposed framework demonstrated an overall 97.29% accuracy.

E. Comparison With Existing Methods

The performance of the proposed LNN-HGNN-EDL framework can be compared with the performance of some of the commonly used machine learning and deep learning techniques for detecting cyber threat in the SCADA/OT environment, which include Random Forest, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Convolutional Neural Network (CNN), and Graph Neural Network (GNN).

ALGORITHM: PROPOSED LNN-HGNN-EDL-BASED CYBER THREAT DETECTION ALGORITHM

Algorithm 1: LNN-HGNN-EDL-Based Cyber Threat Detection

1. Input the SCADA/OT dataset D .
 2. Perform data preprocessing
 3. Divide the processed dataset into training and testing sets.
 4. Extract behavioral features from the SCADA/OT observations.
 5. Construct temporal input sequences from the behavioral.
 6. Initialize the Liquid Neural Network (LNN).
 7. Feed the temporal sequences into the LNN.
 8. Generate temporal behavioral representation:
 $hLNN = LNN(X_t)$
 9. Construct a hypergraph using the relevant SCADA/OT.
 10. Initialize the Hypergraph Neural Network (HGNN).
 11. Propagate information through the hypergraph.
 12. Generate relational representation:
 $hHGNN = HGNN(GH)$
 13. Combine the temporal and relational representations:
 $z = \text{Combine}(hLNN, hHGNN)$
-

14. Pass the integrated representation z
15. Generate class-specific evidence:
 $ek = \text{ReLU}(W_k z + bk)$
16. Calculate the evidential parameters:
 $\alpha_k = ek + 1$
17. Calculate the total evidence:
 $S = \sum \alpha_k$
18. Calculate class-wise prediction:
 $pk = \alpha_k / S$
19. Estimate predictive uncertainty:
 $U = K / S$
20. $Y = \text{argmax}(pk)$
21. Evaluate the predictions using
22. Return the predicted threat class Y ,
End Algorithm

V. SECURITY AND PRACTICAL IMPLICATIONS

The proposed LNN-HGNN-EDL approach can help perform security monitoring in OT environments utilizing SCADA systems by evaluating behavioral patterns and detecting behaviors that are not consistent with the expected system behavior. LNN allows taking into account the changes in behavior and operational state over time whereas HGNN helps to represent the relationships between devices and communication entities. In terms of practicality, the ability of the proposed framework to perform in real-time monitoring depends on a number of aspects like detection time, computational power, processing capabilities, and observation refresh rate. Continuous monitoring helps detect the suspicious behavioral change in real-time. At the same time, it is necessary to take into account the computational needs of the neural networks in choosing the environment for the implementation of the proposed approach. False positive is another important issue that needs to be taken into account in an industrial system. Too many wrong alerts might increase the amount of work for security personnel and distract from their other tasks. The EDL allows getting some extra information in terms of the evidence and uncertainty of the prediction. This information can help treat the uncertain predictions and events differently from those that have stronger evidence.

VI. RESULT AND DISCUSSION

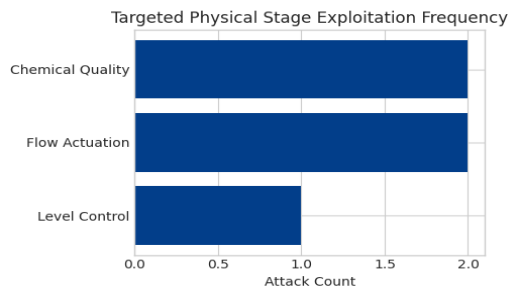


Fig. 2. Targeted Physical Stage Exploitation Frequency

Fig. 2 gives the number of attacks against various physical stages of the industrial process. Each of the Chemical Quality and Flow Actuation stages was subjected to an attack count of 2, making them the most attacked

physical stages in the data set under consideration. On the other hand, the Level Control stage had an attack count of 1. It thus becomes evident from the Fig. below that the number of attacks was not evenly distributed among the physical stages since Chemical Quality and Flow Actuation stages received double the number of attacks of Level Control stage.

TABLE I. TARGETED PHYSICAL STAGE EXPLOITATION FREQUENCY

Physical Stage	Attack Count
Chemical Quality	2
Flow Actuation	2
Level Control	1

TABLE I Summarizes the number of attacks present. Chemical Quality and Flow Actuation are the stages with the most number of attacks, having 2 attacks each, while the Level Control has only 1 attack. It can be inferred from the findings that the exploitation attacks seen were more focused in Chemical Quality and Flow Actuation stages than the Level Control stage. This can help in pinpointing physical process areas for monitoring SCADA/OT security.

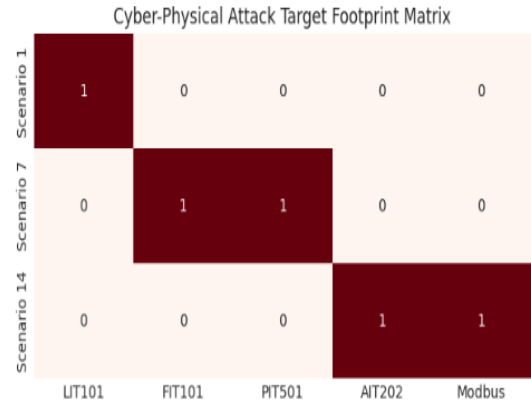


Fig. 3. Cyber-Physical Attack Target Footprint Matrix

Fig. 3 shows how the targets for attacks on cyber-physical systems vary depending on scenarios and industrial assets. For Scenario 1, the attack target is LIT101, and for Scenario 7, it is FIT101 and PIT501. Scenario 14 involves an attack target of AIT202 and Modbus. From the matrix, it can be noted that the target varies based on the scenario and not all monitored assets at once.

TABLE II. CYBER-PHYSICAL ATTACK TARGET FOOTPRINT MATRIX

Scenario	LIT101	FIT101	PIT501	AIT202
Scenario 1	1	0	0	0
Scenario 7	0	1	1	0

Scenario 14	0	0	0	1
-------------	---	---	---	---

TABLE II represents the numeric depiction of the attack-target relationships. A value of 1 means that the asset or communication part is under attack in the corresponding scenario while 0 means there is no attack-target relationship in that case. Scenario 1 includes LIT101, Scenario 7 comprises FIT101 and PIT501, and Scenario 14 includes AIT202 and Modbus. The table provides a structured view of the attack footprint and can be utilized for analyzing the SCADA/OT assets involved in certain cyber-physical attack scenarios.

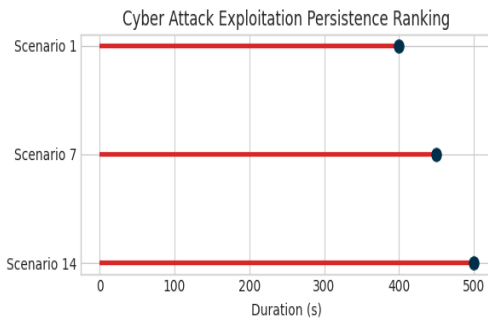


Fig. 4. Cyber Attack Exploitation Persistence Ranking

The persistence time (seconds) of three scenarios of cyber attack exploitation. As shown in Fig. 4, Scenario 1 has persistence of about 400 seconds, Scenario 7 has persistence of about 450 seconds, while Scenario 14 has the highest persistence of 500 seconds.

TABLE III. CYBER ATTACK EXPLOITATION PERSISTENCE RANKING DATA

Scenario	Duration (s)
Scenario 1	400
Scenario 7	450
Scenario 14	500

The numeric analysis of the duration of persistence periods portrayed in the associated graph. From TABLE III, it can be seen that Scenario 1 had a persistence period of exploitation lasting 400 seconds, Scenario 7 achieved 450 seconds, and Scenario 14 had the longest period of 500 seconds.

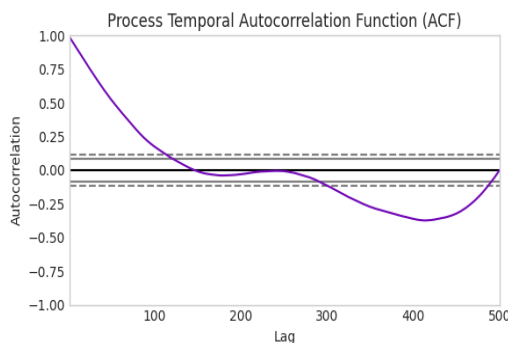


Fig. 5. Process Temporal Autocorrelation Function (ACF)

Fig. 5 provides an example of how time dependence decreases as a function of higher values of lag in the process being observed. The line starts at full positive correlation (1.00) at zero lag, gradually descending as the lag increases. At about 150 lag value, the line moves below the zero level into slightly negative correlation, remaining around the zero level until about lag 250.

TABLE IV. PROCESS TEMPORAL AUTOCORRELATION FUNCTION (ACF) DATA

Lag Interval	Peak	Autocorrelation Trend Description
0	1.00	self-correlation
150	0.00	Initial zero-crossing
200 – 250	~0.00	near zero
410	-0.37	Maximum negative

Identifies important stages of transitions that occurred on the autocorrelation path. As seen in TABLE IV, the process starts at lag 0 showing a totally dependent relationship that gradually drops down to 0 at lag 150. After a short stabilization stage between lags 200 and 250, an inverse correlation occurs at lag 410 having a coefficient of about -0.37.

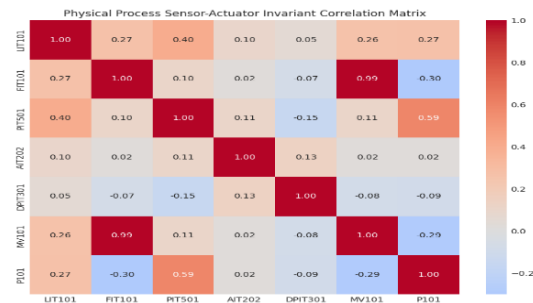


Fig. 6. Physical Process Sensor-Actuator Invariant Correlation

Fig. 6 shows the correlation coefficients between the physical process sensors and actuators. Positive correlations exist between PIT501 and P101 at 0.59 as well as LIT101 and PIT501 at 0.40. There exist inverse relationships between FIT101 and P101 at -0.30 and MV101 and P101 at -0.29. The other node-pairs have weak positive linear correlation and are almost equal to zero.

TABLE V. PHYSICAL PROCESS SENSOR-ACTUATOR INVARIANT CORRELATION MATRIX DATA

Sensor	Correlation Value	Relationship Type
FIT101 & MV101	0.99	Positive
PIT501 & P101	0.59	Positive
FIT101 & P101	-0.30	Moderate
AIT202 & MV101	0.02	Neutral

The actual numerical values of matrix for each sensor actuator pair. As shown in TABLE V, FIT101 and MV101 show the highest value of correlation (0.99), while DPIT301 has very weak interaction with most of the system variables ranging from -0.15 to 0.13.

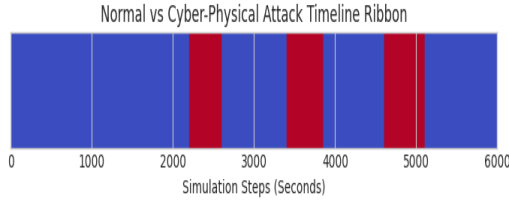


Fig. 7. Normal vs Cyber-Physical Attack Timeline Ribbon

Fig. 7 shows the temporal order in which the system is operating, with periods of cyber-physical attack taking place intermittently during the 6000-second simulation. The blue bars denote intervals when the baseline system is operating normally, while the red bars indicate the time periods during which there are attacks on the system.

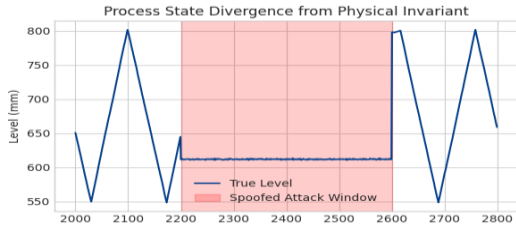


Fig. 8. Process State Divergence from Physical Invariant

Fig. 8 Shows how the true level of the fluid (measured in mm) behaves before, during, and after the spoofed cyber-attack period. Outside the spoofed cyber-attack period, the true level follows a cyclic wave pattern, rapidly fluctuating from 550 mm to 800 mm. But within the red-shaded region representing the spoofed attack period (from 2200 s to 2600 s), the level is kept artificially constant at 612 mm. As soon as the spoofed attack ends at 2600 s, the level jumps up to 800 mm and again follows its usual cyclic pattern.

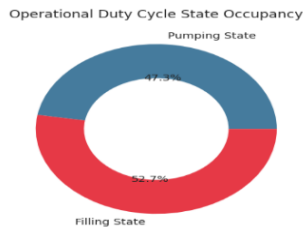


Fig. 9. Process State Divergence from Physical Invariant

Fig. 9 demonstrates the behavior of the actual fluid level (in mm) prior to, during, and after a simulated cyber-attack period. Prior to and after the simulated attack, the actual level behaves in a predictable cyclical fashion, rapidly fluctuating between the minima of 550 mm and maxima of 800 mm.

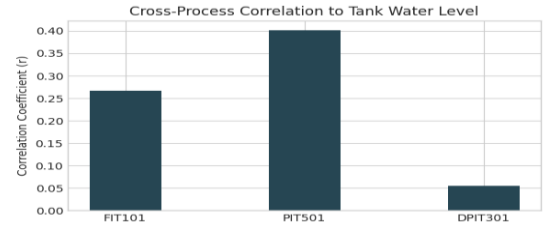


Fig. 10. Cross-Process Correlation to Tank Water Level

Fig. 10 illustrates the linear correlation coefficients (r) between each physical sensor and the tank water level. Of all three process units measured, the highest linear correlation is found for PIT501, with a value of 0.40. FIT101 has a moderately positive correlation of about 0.27, while DPIT301 has the lowest linear correlation of 0.05.

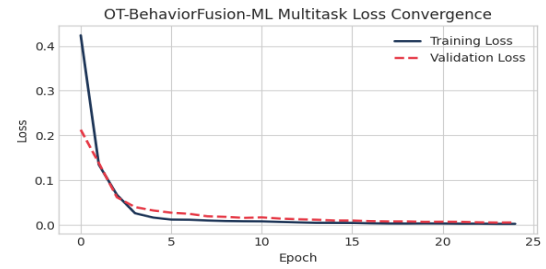


Fig. 11. OT-BehaviorFusion-ML Multitask Loss Convergence

Fig. 11 Shows the loss convergence graphs for training and validation over 25 epochs for the OT-BehaviorFusion-ML multitasking optimization algorithm. The two graphs show a sharp decline between epoch 0 and epoch 3, where the training loss decreases from 0.42 to below 0.03, while the validation loss decreases from about 0.21 to 0.04. From epoch 5 onwards, the two graphs start to level off sharply, converging to zero steadily up to epoch 24.

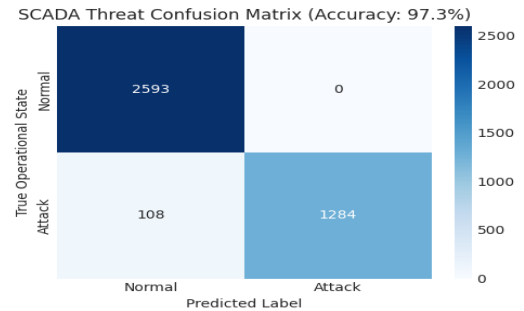


Fig. 12. SCADA Threat Confusion Matrix

Fig. 12 illustrates the classification performance of the model using a confusion matrix across normal and attack operational states, achieving an overall accuracy of 97.3%. Out of 2,593 actual normal operational instances, the model correctly identified all 2,593 samples with zero false positives. For the attack state, the model correctly classified 1,284 attack events while misclassifying 108 attack instances as normal.

OT-BEHAVIORFUSION-ML SCADA THREAT DETECTION REPORT:

	precision	recall	f1-score	support
Normal SCADA Behavior	0.9600	1.0000	0.9796	2593
Cyber-Physical Threat	1.0000	0.9224	0.9596	1392
accuracy			0.9729	3985
macro avg	0.9800	0.9612	0.9696	3985
weighted avg	0.9740	0.9729	0.9726	3985

Fig. 13. OT-behaviorfusion-ML scada threat detection report

Fig. 13 depicts the classification metrics report for the OT-BehaviorFusion-ML model across normal SCADA behavior and cyber-physical threat classes over a evaluation dataset of 3,985 total samples. Normal SCADA behavior achieved a precision of 0.9600, perfect recall of 1.0000, and an F1-score of 0.9796 across 2,593 instances. Cyber-Physical Threat detection recorded perfect precision at 1.0000, a recall of 0.9224, and an F1-score of 0.9596 over 1,392 instances. Overall system accuracy reached 0.9729, with a macro-averaged F1-score of 0.9696 and a weighted-averaged F1-score of 0.9726.

VII. CONCLUSION

This study has developed a machine learning based behavioral analytics model for detection of cyber threats in the context of SCADA based Operational Technology (OT) system that utilizes Liquid Neural Networks (LNNs), Hypergraph Neural Networks (HGNNs), and Deep Evidential Learning (EDL). This behavioral analytics framework caters to the challenges of OT security through its ability to capture the behavioral changes over time, the high order relationships among the devices, communications activities as well as prediction uncertainty in classification of threats. The combination of these capabilities has made it possible for the framework to identify normal behavior of the system from any malicious activity and also provide greater assurance to the decision-makers on safety and security. The results obtained from the experiment have been able to detect the cyber threat patterns with accuracy of 97.29% in the selected SCADA-based OT environment. The proposed methodology also offers an adaptable framework to cope with any future threats and behavior changes in industrial systems. Incorporation of the uncertainty measurement allows the security operator to discern clear predictions from possible uncertain scenarios. The two together could help detect anomalies in a timely manner and avoid unnecessary security actions in important OT environments. Future research may consider implementation in real time, large-scale SCADA data, and improvement of the proposed method. Further research could be centered around testing the proposed model with a larger and more diverse dataset taken from actual SCADA and industrial control systems. In addition, the model may be fine-tuned to be used in practice in a real-time setting where threats are detected in a timely manner and there is constant monitoring of behavior. The model may be improved in order to use adaptive learning approaches to detect new cyber attacks. The model may also be integrated

into security monitoring frameworks and utilize techniques of explainable AI to help security practitioners interpret threat predictions.

REFERENCE

- [1] Al-Qirim, Nabeel, et al. "Cyber threat intelligence for smart grids using knowledge graphs, digital twins, and hybrid machine learning in SCADA networks." *International Journal of Engineering Business Management* 17 (2025): 18479790251328183.
- [2] Uzzaman, Arfan, and M. A. Rony. "Machine Learning-Based Cybersecurity Models for Safeguarding Industrial Automation And Critical Infrastructure Systems." *International Journal of Scientific Interdisciplinary Research* 4.4 (2023): 224-264.
- [3] Kabir, S. Rayhan, et al. "SCADA Digital Twin and Cryptographic AMI Cyber-Twin for Smart Grid Security." *Applied Sciences* 16.15 (2026): 7655.
- [4] Hasan, Mohammad Kamrul, et al. "Randomization based shallow and federated-deep learning for smart grid security using label-encoded vulnerabilities and distributed LSTM computation." *PeerJ Computer Science* 12 (2026): e3354.
- [5] Lu, Kang-Di, et al. "Multi-objective discrete extremal optimization of variable-length blocks-based CNN by joint NAS and HPO for intrusion detection in IIoT." *IEEE Transactions on Dependable and Secure Computing* 22.4 (2025): 4266-4283.
- [6] Sharmeen, Shaila, et al. "An advanced boundary protection control for the smart water network using semisupervised and deep learning approaches." *IEEE Internet of Things Journal* 9.10 (2021): 7298-7310.
- [7] Ramesh, SS Subashka, et al. "EFFECTIVE TIME-SERIES ANOMALY DETECTION MODEL FOR INDUSTRY CONTROL SYSTEM USING IOT AND MACHINE LEARNING TECHNIQUES." *Telecommunications and Radio Engineering* 85.3 (2026).
- [8] PRASANNA, R., et al. "Effective Time-Series Anomaly Detection Model for Industry Control System Using IoT and Machine Learning Techniques." *Telecommunications and Radio Engineering*.
- [9] T. Dinesh, T. Aashika., R. Divya, "AITIR: An Intelligent Cyber Security Framework for Government Agencies," *Journal of Engineering Strategy and Innovation (JESI)*, vol. 1, no. 1, pp. 57–64, 2026.
- [10] Lo, Carol, et al. "Digital twins of cyber physical systems in smart manufacturing for threat simulation and detection with deep learning for time series classification." *2024 29th International Conference on Automation and Computing (ICAC)*. IEEE, 2024.
- [11] Althobaiti, Ahlam Saud. *On Energy Theft Attack Detection in Smart Grids Using Machine Learning*. Diss. Lancaster University (United Kingdom), 2023.
- [12] Yeboah-Ofori, Abel, et al. "Cyber threat predictive analytics for improving cyber supply chain security." *Ieee access* 9 (2021): 94318-94337.
- [13] Mirmarghabi, Mohammad, Ahmad Afshar, and Hajar Atrianfar. "Intrusion Detection System for Securing Agriculture 4.0 Against DDoS Attacks Using Deep Learning and Machine Learning Models." *2025 33rd International Conference on Electrical Engineering (ICEE)*. IEEE, 2025.
- [14] Rehman, Zubaidda, et al. "Machine learning and internet of things applications in enterprise architectures: Solutions, challenges, and open issues." *Expert Systems* 41.1 (2024): e13467.
- [15] Chen, Zhiyan, et al. "Machine learning-enabled iot security: Open issues and challenges under advanced persistent threats." *ACM Computing Surveys* 55.5 (2022): 1-37.
- [16] Morolong, Mamoqenelo, Attlee Gamundani, and Fungai Bhunu Shava. "Investigation of machine learning-inspired security controls used in protecting health critical infrastructure from advanced persistent threat

attacks." 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC). IEEE, 2024.