Anomaly Detection in Cyber-Physical Systems Using Long-Short Term Memory Autoencoders: A Case Study with Man-in-the-Middle (MiTM) Attack

Shining Sun, *Student Member, IEEE,* Khandaker Akramul Haque, *Student Member, IEEE*, Xiang Huo, *Member, IEEE,* Abhijeet Sahu, *Member, IEEE,* Ana Goulart, *Member, IEEE,* Katherine Davis, *Senior Member, IEEE*

***Abstract*—Large-scale power cyber-physical systems (CPSs) have many factors that contribute to uncertainties in their data. When intrusions occur, they will cause anomalies in the system's cyber-physical data. However, the traditional anomaly detection methods often rely on static thresholds or simple statistical models which are not accurate enough to identify the outlier, leading to a higher risk of false positives or missed detections. Recent advances in deep-learning based detection of stealth false data injection attacks offers a number of improvements, but the cohesive use of cyber-physical time domain data from real world systems to detect and validate the detection with a ground truth model from an emulation testbed and their incorporation in real world energy management systems remains in its infancy. Hence, this paper aims to model and capture temporal dependencies with emulation data, enabling unsupervised anomaly detection by reconstructing expected behavior and identifying deviations that suggest potential attacks as recent methods have fallen short in identifying subtle, long-term dependencies. This study proposes a Long Short-Term Memory (LSTM) autoencoder-based approach to detect Man-in-the-Middle (MiTM) attacks in power systems by leveraging multi-sensor temporal datasets [1]. Additionally, feature reduction and data normalization techniques are implemented to improve model performance. Simulations using a Texas 2000-bus grid case demonstrate the effectiveness of our approach in identifying and mitigating cyber threats, effectively enhancing intrusion detection capabilities.**

***Index Terms*—anomaly detection, LSTM autoencoder, cyber threats, multi-sensor data fusion, power grid resilience**

I. Introduction

According to the U.S. Department of Energy (DOE), the number of intrusions has grown sharply in recent years, which emphasizes the urgent need for security-oriented design of engineered systems. Cyber-informed engineering solutions are needed, including intrusion prevention, detection, and response techniques that leverage the physics of the system in an integrated manner with cyber. [2]. Therefore, it is essential to enhance situational awareness and adopt proactive responses related to cyber security issues.

The authors would like to acknowledge the US Department of Energy under award DE-CR0000018, the National Science Foundation under Grant 2220347 and TEES Smart Grid Center.

979-8-3503-3120-2/24/$31.00 ©2024 IEEE

Cyber-physical systems (CPSs), which integrate physical devices with cyber components, aim to enhance the resilience of power systems by providing new theoretical and practically deployed ways of system vulnerabilities and providing proactive responses [3], [4]. However, a wide range of cyber threats pose significant risks to critical cyber-physical infrastructures like power grids [5], which could cause communication channel disruption, power system outages, etc. These threats encompass various types of attacks, such as phishing, denial-of-service (DoS), ransomware, man-in-the-middle (MiTM), and insider threats. The MiTM attack is a type of cyber attack where an intruder secretly intercepts and potentially alters the communication between two ports that believe they are directly communicating with each other [6]. In the MiTM attack, the intruder positions themselves between the sender and receiver, capturing the data being exchanged.

In response to cyber attacks threaten CPSs, anomaly detection is a crucial aspect of ensuring the integrity and security of the system [7], [8]. Traditional intrusion detection systems like Snort may cause false positive results because they rely on predefined rules and patterns. It matches incoming network traffic against its database of known attack signatures and triggers an alert when a match is found, which may cause false positives according to the rules. Machine learning models like *random forest (RF)*, *Support Vector Classifier (SVC)*, can handle static data well but struggle with the temporal structure in sequential data [9], [10]. However, an LSTM autoencoder-based anomaly detection approach enhances system resilience and security by leveraging multi-sensor data to reduce false positives compared to traditional methods. It has demonstrated promising results in outlier identification in various areas, like air quality prediction [11], discovering suspicious vehicle network activities [12], etc. The LSTM autoencoder is designed to model the temporal dependencies, reflecting the dynamic interactions within the cyber and physical components making them ideal for capturing the sequential and time-dependent nature of CPS data. This approach allows for more precise identification of cyber threats, ensuring timely and accurate responses to potential cyber intrusions.

In this paper, we develop an LSTM autoencoder-based approach to detect MiTM attacks on a 2000-bus grid cyber-

2025 IEEE Texas Power and Energy Conference (TPEC) | 979-8-3315-4112-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/TPEC63981.2025.10906975

physical datasets [1]. The proposed detector can effectively identify anomalies within the data with improved detection precision and accuracy. Moreover, feature reduction minimizes irrelevant or redundant information, and data normalization scales the features, allowing the model to focus on significant patterns and achieve faster convergence during training which optimize the model's performance.

The subsequent sections of the paper are structured as follows: Section II presents a comprehensive overview of the case study. Section III elaborates on the method design. This is followed by an in-depth examination of the results in Section IV. Finally, Section V provides concluding remarks.

II. Related Work and Literature Review

A. Multi-Source Multi-Domain Data Fusion during MiTM Attack

The RESLab testbed provides a real-time cyber-physical platform to show the interaction between cyber and physical systems. It could continuously monitor network traffic and mimic real-world system activity. It consists of a network emulator, the dynamic power system simulator, the intrusion detection system (IDS), a Real-Time Automation Controller (RTAC), the data storage and fusion system, and the OpenDNP3 master [6], [13]. The Common Open Research Emulator (CORE) serves as the network emulator, which could model, simulate, emulate, test, and validate the monitoring system while Power World Dynamic Studio (PWDS) simulates power systems in real-time. The DNP3 Master, implemented using an open DNP3-based application and an SEL-3530 RTAC, manages communications with the power system, polls measurements, and operates outstations within the simulated environment. SNORT is implemented to configure and generate alerts for DoS, MiTM, and Address Resolution Protocol (APR) attacks. More demonstrations are illustrated in [6], [13].

The fusion engine in the RESLab testbed enables the capability to collect data from multiple sensors, synthesizing real-world data in the energy management system (EMS) [6], [13]. For cyber sensors, Wireshark instances capture raw packets at various network locations, and Packetbeat is used to extract network flow-based data. Security monitoring includes Snort IDS logs and alerts. For the emulated physical system using PWDS, real-time sensor readings are obtained from the observed measurements at the DNP3 master, based on the raw packet data captured at the DNP3 master. The multivariate time series data is implemented as our input to the LSTM autoencoder-based approach framework.

The dataset includes simulations of several cyber attacks conducted in the RESLab testbed [14]. The MiTM attacks simulated in the RESLab emulate multi-stage attacks on a synthetic electric grid, where an intruder gains Secure Shell (SSH) access and performs coordinated False Command Injection (FCI) and False Data Injection (FDI) attacks, accomplishing ARP spoofing, to overload transmission lines by compromising DNP3 communications. Detailed scenarios are discussed in section IV-A2 and the referenced testbed and MiTM attack papers [6], [13], [15].

B. Anomaly Detection Literature Review

Anomaly (or outlier) detection focuses on two categories of data: erroneous or unwanted data and the data under certain events [16]. In this paper, anomalies are defined as data distinct from the power system's normal operation state. Anomalies occur when data significantly diverges from the values it holds over a period of time, such as a sudden drop in physical data caused by MiTM attacks. .

Our previous research [13] explored machine learning techniques including supervised learning approaches unsupervised learning techniques and semi-supervised learning, which also demonstrated the capability of outlier identification facing cyber attacks. However, traditional machine learning techniques often struggle with capturing the temporal dependencies inherent in time series data. LSTM, as an extension of recurrent neural networks (RNNs), effectively addresses the issue of short-term memory by maintaining information over longer sequences [11], [12]. LSTM autoencoder, where the LSTM cell captures the dependencies and autoencoder reconstructs the sequence to identify the anomalies, making them particularly suitable for time series prediction tasks [17]. For example, [18] utilized the LSTM-autoencoder model for studying temporal correlations between the feature vectors extracted from the state estimation during a false data injection attack in power systems. Since LSTM autoencoders can train sequential data, it is worthwhile to investigate how to leverage this technique to capture the interdependencies between cyber and physical data so as to indicate the abnormal activity and enhance the detection capability.

III. Methodology

The framework of the proposed LSTM autoencoder-based anomaly detection method is shown in Fig.1. Specifically, it

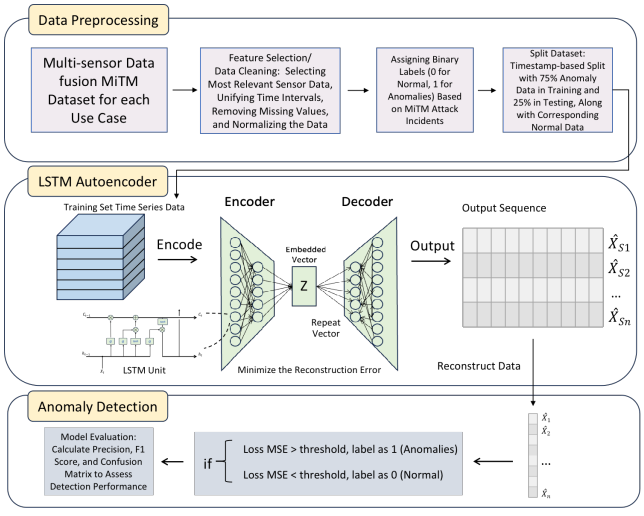


Fig. 1: LSTM Autoencoder-Based Detection of Multivariate Time Series Data for Cyber-Physical Power Systems

consists of three steps, including data preprocessing, model

training, and anomaly detection.Initially, the data preprocessing step involves feature selection, data cleaning, and the labeling of datasets, ensuring that only relevant and high-quality data is used for model training. The dataset is then split into training and testing sets to facilitate effective model evaluation. Next, an LSTM autoencoder is employed, where the input data is encoded into a lower-dimensional latent representation (z) that captures essential features of the time-series data. The decoder reconstructs the original data from this latent space, with the objective of minimizing the reconstruction error for normal data sequences. Finally, the anomaly detection step reconstructs the feature variables and computes the reconstruction error, measured by the Mean Square Error (MSE), to identify anomalies. Data points with a reconstruction error exceeding the threshold are labeled as anomalies, while those with lower errors are considered normal. The details of the three steps in the framework are discussed as follows.

A. Data Preprocessing

Data preprocessing is essential to ensure the quality of the training and testing datasets as well as the integrity of the subsequent LSTM autoencoder model training. The original dataset is not in a unified time interval, so the resampling method is applied to ensure consistency in the time intervals across data from the DNP3 master, router, and DNP3 outstation.This resampling was crucial for aligning and integrating the temporal data.

The dataset initially includes 28 cyber features along with physical data (power flows) obtained from DNP3. Among which, 14 features are selected as the most relevant based on *principal component analysis (PCA)* analysis [13], including critical physical variables such as voltage and current, as well as other cyber sensors that can indicate the system's behavior.

Datasets are encoded because the features in the dataset are categorical. Label encoding is implemented by the Scikit-learn library. It is preferred since it avoids the high-dimensionality issues that can arise with one-hot encoding [13].

Moreover, the dataset is scaled using Eq. (1). By scaling the features to a uniform range within [0,1], it reduces the risk of any single feature dominating the learning process and improves the feature processing efficiency [19].

$$X' = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \tag{1}$$

The data is then split into training and testing sets. The split is managed in a way that can ensure that both training and testing sets contain a sufficient number of anomaly data points. The dataset is split based on timestamps, with the training set containing 75% of the anomaly data and the testing set containing the remaining 25%, along with the corresponding normal data.

Anomaly data labeling is another crucial aspect of the preprocessing phase. Anomalies can be labeled during known cyber attacks, as well as during periods when the voltage and current measurements indicated instability (e.g., values falling below normal operational thresholds).

B. LSTM Autoencoder

1) LSTM Cell: The LSTM cell state is updated at each time step, allowing the model to remember or forget information as needed [20], described by:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{2}$$

where C_t is the cell state at time step t, f_t is the forget gate, i_t is the input gate, and $\tilde{C}_t$ is the candidate cell state.

Specifically, three gates manage and control the information flow determined using long-term and short-term memory [20]. In the following equations, W denotes the weight matrix, and b denoted the bias term:

- Forget Gate: It determines the percentage of information to erase from the cell state. A sigmoid activation function is implemented as described by Eq. 3. The activation function takes the previous hidden state and the current input to output a value between 0 and 1 for each iteration in the cell state.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$

- Input Gate: It controls new information to add to the cell state. As Eq. 4 illustrates, the sigmoid function determines the values to update, and a $\tanh$ layer decides a vector of new values to be added, shown in Eq. 5.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{5}$$

- Output Gate: By combining the current state and the hidden state values (Eq. 6), it further determines the output value by multiplying Eq. 6 with a $\tanh$ layer into Eq. 7 as:

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

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

2) Autoencoder: The autoencoder consists of an encoder and a decoder to reconstruct the input updating the weights (W) and bises (b) during training. The encoder maps the input feature spaces x to the latent space z which is shown in Eq. 8. The activation function ϕ transforms the input into the latent space, often referred to as the bottleneck [21]. The decoder reconstructs the output x' from z using a neural network, demonstrated in Eq. 9.

$$z = \phi(W_e \cdot x + b_e) \tag{8}$$

$$x' = \psi(W_d \cdot z + b_d) \tag{9}$$

The Mean Squared Error (MSE)is calculated to quantify the differences between the input x and the reconstructed x' by:

$$\mathcal{L}(x, x') = \frac{1}{n}\sum_{i=1}^{n}(x_i - x'_i)^2 \tag{10}$$

Therefore, the training process optimizes the model by minimizing the discrepancy between the input x and the reconstructed x'.

C. Anomaly Detection

Once the model is trained, the training dataset is passed through the model to evaluate its reconstruction performance and to determine the threshold. The model recreates the original input sequences, and the reconstruction error is calculated for each data point. The threshold for anomaly detection on the test dataset is then established, typically by analyzing the distribution of reconstruction errors in the training data.

The model generates a predicted dataset which is the reconstructed sequences $\hat{\mathbf{X}}_{\text{test}}$, containing multiple reconstructed $\hat{\mathbf{x}}_{\text{test},t}$ for different time t resulting from the application of the sliding window technique. For each time step, the final predicted value $\hat{\mathbf{x}}_{\text{test},t}$ is computed by averaging the predictions from all windows. This can be expressed as:

$$\hat{\mathbf{x}}_{\text{test},t} = \frac{\sum_{i=1}^{N} \hat{\mathbf{X}}_{\text{test}}^{(i)}(t-i+1)}{n_t} \tag{11}$$

where N is the total number of windows that cover time step t, and n_t is the number of windows contributing to the reconstruction. Eq. (11) ensures that the reconstructed value at each time step is averaged based on the number of windows that overlap with it.

Anomalies are identified by comparing the fully reconstructed values $\hat{\mathbf{x}}_{\text{test},t}$ with the original test data $\mathbf{x}_{\text{test},t}$. Significant deviations measured by the reconstruction error (MSE) indicate points where the model fails to capture the underlying patterns. High reconstruction error suggests the potential presence of an anomaly, as these points deviate from the normal behavior learned during the training phase.

IV. Result and Discussion

A. Simulation in Details

1) Data Description: Fused multi-sensor real-time data is acquired from both the physical and cyber sensors. As shown in Fig. 2, cyber data include frame length (frame_len), TCP length (tcp_len), TCP round trip (tcp_rtt), flow count (flow_count), packets, DNP3 Object Count, DNP3 Objects, application layer control count (AL_dnp3_al_ctl), DNP3 application layer object count (AL_dnp3_obj) and DNP3 application layer payload (AL_payload). The physical data is simulated through PWDS with a substation's power flow and injections in branches and buses. There are 5 and 10 DNP3 outstations (os) being polled. The poll rate refers to how frequently the system checks for new data points over a given period. 30poll or 60poll refers to polling intervals of 30 or 60 seconds respectively. The data are resampled at 30-second intervals using the Pandas library, allowing for a uniform time series representation.

2) Attack Scenarios: 4 use cases (UCs) are conducted in the presence of MiTM attacks, see more detailed descriptions in [14].

UC1: The MiTM intrusion employs False Command Injection (FCI) to alter the binary control commands from the RTAC and then modify the commands from the DNP3 master, resulting in CLOSE commands being overridden by TRIP commands. This action opened critical branches, leading to line overloads [13], [14].

UC2: This case implements a scenario where an intruder modifies analog control commands and binary commands to cause power system disruptions. The intruder first inspects DNP3 packets, altering generator set points to zero and changing binary control commands as done in UC1. In this case, seven generators and one transmission branch are compromised [13], [14].

UC3: In this scenario, the intruder manipulates polled measurements, leading the operator to re-send control commands, which involves FDI attack. The intruder then modifies these commands, by altering generator set points.

UC4: It is a three-stage attack where the intruder first alters the polled measurements from the DNP3 master, prompting the operator to re-send control commands. The intruder then changes the generator set points to low values while falsifying measurement packets to show the original set points, deceiving the operator into thinking the commands were successful. As a result, the true generator outputs decrease, risking an overload when a line is opened.

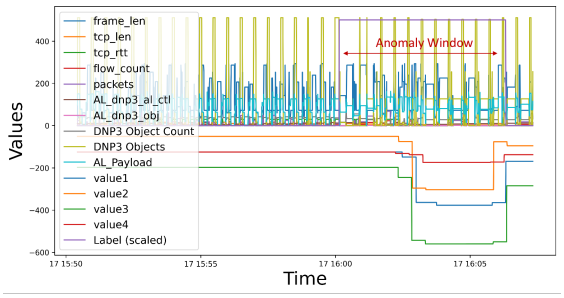
3) Model: The first LSTM layer of the encoder takes the input sequence and outputs a sequence of 128-dimensional vectors for each time step. The second LSTM layer serves as a bottleneck, reducing the output to a 64-dimensional vector. Unlike the first layer, it outputs only a single vector, effectively encoding the entire input sequence into a single 64-dimensional representation. The latent variables in this LSTM autoencoder are represented by the output of this bottleneck layer, which, in this case, has a latent vector dimension of 64. The decoder then takes this compressed representation, replicates it, and reconstructs a sequence that resembles the original input. Dropout layers are employed to prevent overfitting by randomly dropping units during training. Finally, the TimeDistributed layer maps the decoder's output to the desired shape, which in this example is a sequence of 14-dimensional vectors.

4) Evaluation Metrics: F1-score, Recall, and Precision are implemented to evaluate overall performance. Precision indicates the proportion of true positive (TP) divided by the total number of elements labeled positive, including false positive (FP) and TP. Recall is defined as the number of TP divided by the total number of actual positives. Accuracy is the ratio of correctly predicted elements to the total elements. The F1 score is defined as the harmonic mean of the precision and recall, which measures the test accuracy [22].

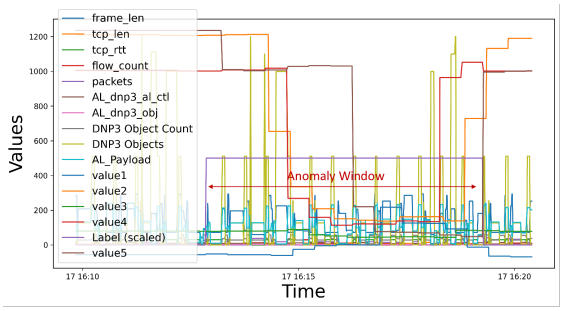
B. Result Analysis and Discussion

In this section, the anomaly detection performance and overall effectiveness are evaluated through 4 UCs.

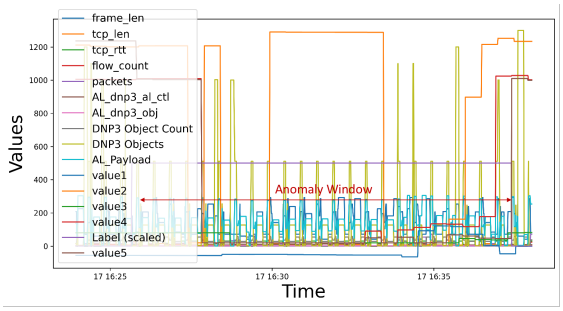
1) Anomaly Detection Performance: As is shown in Fig. 3, the loss MSE line describes the difference between the true value and the predicted value. It suggests that higher differences indicate a higher prediction error, which implies anomalies. The red dashed line is a threshold for anomaly detection. If the loss, MSE, exceeds the threshold, the model



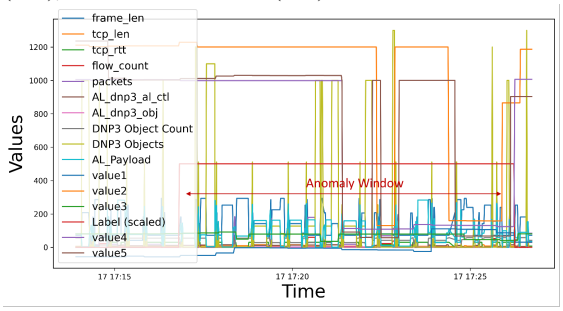
(a) UC1: Overloaded transmission lines (WACO 3 (399), WACO 1 (456), JEWETT 1 (1195), and FRANKLIN (1200)).



(b) UC2: Real Power injection at generators from WADSWORTH(968), RIESEL 1 (631), GRANBURY 1 (601), and GLEN ROSE 1 (560) and the overloaded line.



(c) UC3: Power injection at WADSWORTH generator.



(d) UC4:Power injection at WADSWORTH generator during mixing of FDI and FCI.

Fig. 2: Multivariate Time Series Data of MiTM Attacks across Use Cases UC1-UC4 with a 30-Second Polling Rate. The label in the legend is scaled up to highlight the window of anomalies during cyber attacks and physical disturbances.

will flag the point as an anomaly. The vertical green lines indicate where anomalies are detected by the model. These points correspond to when the loss MSE exceeds the threshold, marking the time steps where the behavior is considered abnormal. The dotted line shows the true labels from the test set, indicating where actual anomalies were present. These true labels are used to evaluate the performance of the model in detecting anomalies. Most anomalies detected by the model align closely with the true labels, demonstrating its effectiveness, though a few false positives and false negatives were observed.

2) Overall Performance: Table I highlights the precision of LSTM autoencoder-based anomaly detection approach and how different polling rates and outstation numbers impact the performance of anomaly detection models across various use cases. A main observation across all use cases is the high precision of the LSTM autoencoder, especially in UC1 and UC4. For example, in UC1 with the 10os_60poll configuration, the model achieves a precision of 0.9854, indicating that most of the detected anomalies are true positives. Similarly, in UC4 with the same configuration, the precision reaches 0.9849, further emphasizing the model's effectiveness in reducing false positives.

However, trade-offs in recall are evident across some use cases. In UC4 with 10os_60poll, although the precision is exceptionally high at 0.9849, the recall drops to 0.6776, leading to a lower F1-Score of 0.8029. This suggests that in certain settings, while the LSTM autoencoder confidently detects anomalies, it may also miss some false negatives.

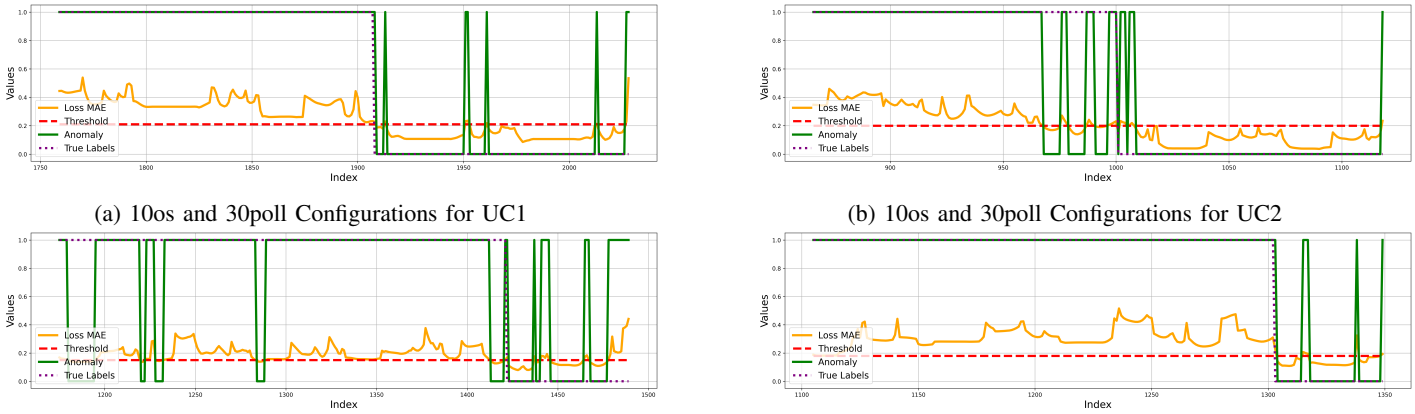
Conversely, UC3 demonstrates how varying outstation numbers and polling rates can significantly impact performance. For instance, in UC3 with 5os_60poll, the recall is notably high at 0.9736, but precision falls to 0.7988, implying that while more anomalies are captured, there may also be an increase in false positives, resulting in a lower overall F1-Score compared to other use cases.

TABLE I: Polling rate and performance metrics for anomaly detection use cases (UC1 to UC4).

UC	Polling Rate	Precision	Recall	F1-Score
UC1	10os_30poll	0.967105	0.980200	0.97351
UC1	10os_60poll	0.985366	0.980583	0.982968
UC2	10os_30poll	0.941667	0.837037	0.886274
UC2	10os_60poll	0.929348	0.994186	0.960674
UC3	10os_30poll	0.901709	0.854251	0.877338
UC3	5os_60poll	0.798762	0.973585	0.877551
UC4	10os_30poll	0.906822	0.974955	0.939655
UC4	10os_60poll	0.984887	0.677643	0.802875

V. Conclusion

The LSTM Autoencoder-based framework can efficiently detect anomalies caused by the MiTM in CPS. Our work examines the relationship between the loss MSE, threshold, and detected anomalies to assess how well the model identifies abnormal behavior within different use cases. The general



(a) 10os and 30poll Configurations for UC1

(b) 10os and 30poll Configurations for UC2

(c) 10os and 30poll Configurations for UC3

(d) 10os and 30poll Configurations for UC4

Fig. 3: Anomaly Detection Performance of UC1-UC4: Comparison of Loss MSE, Threshold, Detected Anomalies, and True Anomaly Labels

performance table is further illustrated to prove the capability to detect the MiTM attack and other disturbances in the CPS. The polling rate and the number of outstations impact the precision of anomaly detection. The real-time reaction is crucial in improving system resilience and security for CPSs, yet it remains a challenging question for both researchers and operators [3]. Future work aims to integrate real-time anomaly detection capabilities, as well as localization techniques, within the RESLab testbed. Localization will enable not only the detection of anomalies but also the identification of the components facing cyber threats, which will enhance the responsiveness of CPSs, providing deeper insights into the real-time detection and mitigation of anomalies.

REFERENCES

[1] A. Sahu, Z. Mao, P. Wlazlo, H. Huang, K. Davis, A. Goulart, and S. Zonouz, "Cyber-physical dataset for mitm attacks in power systems," 2021. [Online]. Available: https://dx.doi.org/10.21227/e4dd-2163

[2] Department of Energy, "Multiyear plan for energy sector cybersecurity," 2018. [Online]. Available: https://www.energy.gov/ceser/articles/doe-multiyear-plan-energy-cybersecurity

[3] A. K. Tyagi and N. Sreenath, "Cyber physical systems: Analyses, challenges and possible solutions," *Internet of Things and Cyber-Physical Systems*, vol. 1, pp. 22–33, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667345221000055

[4] S. Sun, K. A. Haque, X. Huo, L. Homoud Al, S. Hossain-McKenzie, A. Goulart, and K. Davis, "A reinforcement learning engine with reduced action and state space for scalable cyber-physical optimal response," *arXiv preprint arXiv:2410.04518*, 2024.

[5] The White House, "National cybersecurity strategy," 2023, https://www.whitehouse.gov/wp-content/uploads/2023/03/National-Cybersecurity-Strategy-2023.pdf.

[6] P. Wlazlo, A. Sahu, Z. Mao, H. Huang, A. E. Goulart, K. R. Davis, and S. A. Zonouz, "Man-in-the-middle attacks and defense in a power system cyber-physical testbed," *CoRR*, vol. abs/2102.11455, 2021. [Online]. Available: https://arxiv.org/abs/2102.11455

[7] 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 (CSUR)*, vol. 54, no. 5, pp. 1–36, 2021.

[8] C. Mujeeb Ahmed, M. A. Umer, B. S. S. Binte Liyakkathali, M. T. Jilani, and J. Zhou, "Machine learning for cps security: applications, challenges and recommendations," *Machine Intelligence and Big Data Analytics for Cybersecurity Applications*, pp. 397–421, 2021.

[9] Y. Luo, Y. Xiao, L. Cheng, G. Peng, and D. D. Yao, "Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities," 2021. [Online]. Available: https://arxiv.org/abs/2003.13213

[10] I. Imran, S. M. Ali, R. Faiz, M. Alam, S. Imran Ali, M. Bari, and M. Shibli, "A survey of machine learning techniques for detecting anomaly in internet of things (iot)," *Journal of Independent Studies and Research Computing*, vol. 21, 06 2023.

[11] Y. Wei, J. Jang-Jaccard, W. Xu, F. Sabrina, S. Camtepe, and M. Boulic, "Lstm-autoencoder-based anomaly detection for indoor air quality time-series data," *IEEE Sensors Journal*, vol. 23, no. 4, pp. 3787–3800, 2023.

[12] J. Ashraf, A. D. Bakhshi, N. Moustafa, H. Khurshid, A. Javed, and A. Beheshti, "Novel deep learning-enabled lstm autoencoder architecture for discovering anomalous events from intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4507–4518, 2021.

[13] A. Sahu, Z. Mao, P. Wlazlo, H. Huang, K. Davis, A. Goulart, and S. Zonouz, "Multi-source multi-domain data fusion for cyberattack detection in power systems," *IEEE Access*, vol. 9, pp. 119 118–119 138, 2021.

[14] ——, "Cyber-physical dataset for mitm attacks in power systems," 2021. [Online]. Available: https://dx.doi.org/10.21227/e4dd-2163

[15] A. Sahu, P. Wlazlo, Z. Mao, H. Huang, A. Goulart, K. Davis, and S. Zonouz, "Design and evaluation of a cyber-physical testbed for improving attack resilience of power systems," *IET Cyber-Physical Systems: Theory & Applications*, vol. 6, no. 4, pp. 208–227, 2021.

[16] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," 2020. [Online]. Available: https://arxiv.org/abs/2002.04236

[17] S. Githinji and C. W. Maina, "Anomaly detection on time series sensor data using deep lstm-autoencoder," in *2023 IEEE AFRICON*, 2023, pp. 1–6.

[18] L. Yang, Y. Zhai, and Z. Li, "Deep learning for online ac false data injection attack detection in smart grids: An approach using lstm-autoencoder," *Journal of Network and Computer Applications*, vol. 193, p. 103178, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804521001880

[19] V. Gupta and R. Hewett, "Adaptive normalization in streaming data," in *Proceedings of the 3rd International Conference on Big Data Research*, 2019, pp. 12–17.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] D. Bank, N. Koenigstein, and R. Giryes, "Autoencoders," *CoRR*, vol. abs/2003.05991, 2020. [Online]. Available: https://arxiv.org/abs/2003.05991

[22] Y. Sasaki, "The truth of the f-measure," *Teach Tutor Mater*, 01 2007.