Securing Critical Infrastructure: A Denoising Data-Driven Approach for Intrusion Detection in ICS Network

Urslla Uchechi Izuazu, Vivian Ukamaka Ihekoronye, Dong-Seong Kim, Jae Min Lee
Department of IT Convergence Engineering, Kumoh National Institute of Technology, Gumi, South Korea
uursla8@kumoh.ac.kr, ihekoronyevivian@gmail.com, (dskim, ljmpaul)@kumoh.ac.kr

***Abstract*—The centralized and vulnerable nature of the industrial control system (ICS) communication network makes it an attractive target for malicious actors aiming to infiltrate and exploit vulnerabilities. These threat actors seek to cause disruptions, compromise sensitive data, and potentially sabotage critical industrial processes. Existing methods for threat detection assume an ideal scenario where there exists no noise/disturbance to threat detection and classification, neglecting to account for the inherent noise and complexity present in real-world industrial processing environments. In reality, the deployment of these models may introduce performance degradation leading to sub-optimal model performance. In response to the identified issue, this study presents a security framework that proactively addresses the challenges posed by noise and provides a robust mechanism for detecting malicious activities from routine industrial network operations. The proposed framework can be deployed at the supervision network segment of ICS to analyze incoming network traffic signals, to effectively distinguish an attack from normal operation amdist noise. Our proposed approach undergoes experimental simulations to validate its effectiveness, and is compared with state-of-the-art based on key performance metrics. Simulation results show that our approach is robust in reconstructing noisy traffic signals with a minimal mean square error of 0.12 and an overall accuracy of 99.6%, outperforming existing methods.**

***Index Terms*—Autoencoder, Denoising Autoencoder Intrusion Detection, ICS, LSTM, Security**

I. Introduction

The Internet of Things (IoT) connects devices across domains, including industrial systems like power grids and cyber-physical systems [1]. Industrial Internet of Things (IIoT) integrates technologies in manufacturing, driving Industry 4.0 for improved efficiency, quality, and safety [2]. This interconnected ecosystem includes and leverages industrial control systems (ICS), providing an all-encompassing approach to smart industrial operations. A typical ICS comprises components like distributed control systems (DCSs), supervisory control and data acquisition (SCADA) systems, programmable logic controllers (PLCs), human-machine interfaces (HMIs), and sensors, which collectively play crucial roles in overseeing mission-critical control functions across various industrial sectors [3].

The modern ICS architecture consists of 4 key segments as shown in Fig. 1: corporate network, supervision network, production network, and the physical. The corporate network supports critical communication and data exchange, the supervision network processes data and facilitates control commands, the production network enables process control [4], and data transmission from sensors at the physical layer. The integrated setup of ICS enhances operational efficiency but also makes ICS networks attractive targets for threats, given the original design oversight and insecurity of communication protocols. Additionally, the lack of operational technology and software updates further compounds security challenges in the ICS environment.

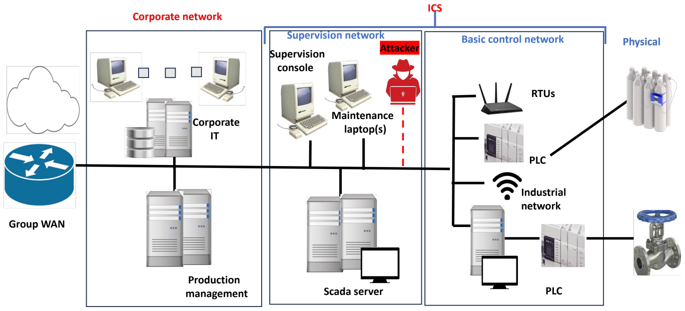

Fig. 1. Industrial Control System Network Architecture

Attacks are usually targeted at the supervisory network as shown in Fig. 1, because it is central in overseeing and coordinating the entire industrial process, hence attacks at this unit can provide adversaries with significant leverage and impact. The 2015 “BlackEnergy 33” incident targeting Ukraine’s power grid exemplifies the vulnerability of ICS to cyber threats [5], emphasizing the global concern for ICS security. Conventional defenses like firewalls and authentication systems have limitations in securing ICS networks due to the diverse protocols within the critical infrastructure, hence the need for artificial intelligence (AI) based intrusion detection systems (IDS).

A. Research Problem Description/Motivation

Conventional machine learning (ML) models for intrusion detection in ICS, exhibit moderate to very low overall performance, which is sub-optimal for time-critical systems like ICS. The solution lies in advanced deep learning (DL) models, renowned for representing intricate and non-linear processes. However, the susceptibility to overfitting in DL models, stemming from their extensive parameter count and noise distortions in real-world industrial settings, hinders their effective deployment for attack detection. Moreover, existing IDS are mainly designed for standard network communication protocols such as TCP/IP [6]. Hence, DL models face challenges in effectively managing control protocols like Modbus and DNP3 [7], which are widely used in critical infrastructures. Additionally, vast feature selection methods have been proposed to generate optimal feature sets for improved performance, but these methods often emphasize statistical properties, overlooking the efficient learning of hidden intrinsic structures within high-dimensional features. In contrast, feature extraction models, such as autoencoders, excel at learning high-dimensional features and compressing them into a set of low-dimensional features that encapsulate the intrinsic core structure of network traffic.

This study introduces a novel framework known as DAE-LSTM, that integrates a denoising autoencoder (DAE) and long short-term memory (LSTM) to handle noisy traffic flow. Thus, addressing real-world challenges with missing or corrupted values emanating from faulty sensor measurements or vibrations in industrial processes. The DAE-LSTM framework effectively removes noise from traffic data, condenses its representation, and performs classification tasks using the dense layer, providing a robust solution for industrial operation perturbations. Our approach contributes significantly to the field, by enhancing the model's feature learning capabilities, crucial for successful attack detection in ICS.

This study makes the following significant contributions:

1) The integration of a denoising autoencoder (DAE); a uniquely configured self-learning feature extraction algorithm with LSTM units, offering a robust security solution for intrusion detection and categorization in the ICS network.
2) The incorporation of a regularization technique to reinforce the model against noise interference. This acknowledges the prevalent noise and disturbances in real-world industrial settings, enhancing the model's resilience and effectiveness in detecting intrusions.
3) The proposed framework is evaluated using the ICS-Flow dataset, chosen for its comprehensive representation of real-world industrial scenarios, after a thorough analysis of its properties.

The paper proceeds as follows: Section II reviews related works. Section III introduces our model. Section IV presents model evaluation and comparison with state-of-the-art. Section V concludes the study and outlines future research plans for IDS in ICS.

II. Related Works and Research Gaps

In the domain of ICS cybersecurity, ML and DL methods have gained wide recognition. Notably, [8] enhanced an LSTM-based framework for anomaly detection. However, their evaluation was based on limited metrics, casting doubt on the robustness of the model. [9] developed an IDS based on the random forest (RF) classifier outperforming other ML classifiers. Yet concerns arose about its real-world applicability due to dataset balancing, as such is not applicable in a real ICS environment. Additionally, [4] designed a combined framework that also addresses the issues with unbalanced data. Authors in [10] proposed an IDS, using chi-square-based feature extraction with a modified decision tree. Their method shows good performance across diverse datasets and is validated by Cohen's kappa coefficient and Mathews correlation coefficient metrics. Also, [11] built an LSTM/AE for intrusion detection in IICS networks which outperforms other models on key metrics. Despite the progress in advancements, a common limitation persists across these models; the oversight of noise impact in industrial settings, which is vital for real-world deployment, posing a challenge to achieving optimal outcomes, resulting in model performance degradation.

A. Basic Concept

1) Learning mechanism for Conventional LSTM: LSTM overcomes vanishing and exploding gradient issues in training long sequences within recurrent neural network (RNN) architecture. Its complex structure revolves around a memory cell and gating mechanisms. These gates, employing the sigmoid function, regulate information in the cell state. The forget gate f_t decides what to retain or delete, the input gate i_t determines information for storage, and the output gate o_t, decides the portion of the cell to output. The calculations for these gates are expressed as equations 1, 2, and 3:

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

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$O_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{3}$$

where

f_t, i_t, O_t :	Activation vectors (output) at time t
σ :	Sigmoid activation function
W_f, W_i, W_o :	Weight matrices for forget, input, and output gates
h_{t-1} :	Hidden state vector at time $t-1$
x_t :	Input vector at time t
b_f, b_i, b_o :	Bias vectors for forget, input, and output gates

B. Learning mechanism for DAE

Autoencoders are self-learning feature extraction algorithms, that play a vital role of extracting crucial patterns from data and compressing it into a lower-dimensional representation. Consisting of an encoder and decoder, the former captures key features, mapping them to a reduced latent space, while the latter reconstructs the original data. A specialized version called the denoising autoencoder (DAE), extends the basic autoencoder by training to remove noise from input signals. DAEs enforce regularization, acquiring robust, noise-free features in the hidden layer. The denoised input is then reconstructed through the decoder. The training aims to minimize the reconstruction error, and it is quantified by the mean squared error (MSE).

While DAE architectures were originally designed for unsupervised learning as feature extraction techniques to eliminate noise from input data [12], [13], their structure can be effectively repurposed for supervised learning tasks, as demonstrated in this study. The fundamental operation of DAE can be represented as equation 4:

$$\dot{\mathbf{X}} = g\left(f(\tilde{\mathbf{X}};\theta)\right) \tag{4}$$

where $\dot{\mathbf{X}}$ is the corrupted input, f, and g are the encoding and decoding functions, parameterized by θ. The objective is to minimize the reconstruction loss presented as equation 5:

$$L(\theta) = \sum_{i=1}^{M} \|\mathbf{X} - \dot{\mathbf{X}}\|_2^2 = \sum_{i=1}^{M} \|\mathbf{X} - g(f(\tilde{\mathbf{X}};\theta))\|_2^2 \tag{5}$$

where $L(\theta)$ is the loss function, $\mathbf{X}$ is the original data, $\tilde{\mathbf{X}}$ is the reconstructed output, and the index i represents each data point.

C. Gaussian noise

Gaussian noise represents a type of signal noise, ranging from 0.1-0.4, characterized by a probability density function, that mirrors that of the normal distribution, commonly referred to as the “Gaussian distribution” [14]. Implying that the possible values that a noise may assume follow a Gaussian distribution. The probability density function p for a Gaussian random variable z can be expressed as equation 6:

$$p(z) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(z-\mu)^2}{2\sigma^2}\right) \tag{6}$$

Here, z denotes the grey level, μ represents the mean grey value, and σ stands for its standard deviation.

III. System Model

Fig. 2, shows our proposed scheme’s workflow with four stages: Preprocessing and noise injection, DAE-LSTM training, LSTM Dense training, and model evaluation.

A. Preprocessing Stage

In this stage, the raw network traffic signal is preprocessed. It involves encoding labels, interpolating, and standardizing the data. Encoding ensures proper representation of the target variable, interpolation was initiated to avoid data loss or leakage, and standardization scales the datasets to a format suitable for model use. The dataset is then split into an 80% training and 20% testing set, after the incorporation of a noise value of 0.1. The strategic inclusion of noise was inspired by real-world scenarios in industrial control systems, where sensor measurements may be tainted by noise or exhibit undesired behavior, posing challenges in accurately detecting instances of attacks, resulting in performance degradation.

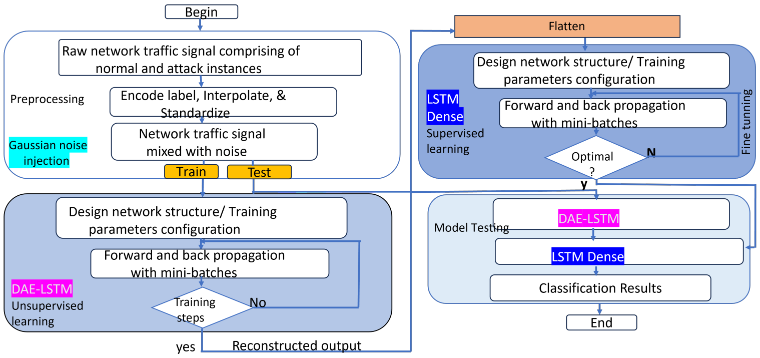
Begin
Raw network traffic signal comprising of normal and attack instances
Preprocessing
Encode label, Interpolate, & Standardize
Gaussian noise injection
Network traffic signal mixed with noise
Train
Test
Design network structure/ Training parameters configuration
Forward and back propagation with mini-batches
DAE-LSTM
Unsupervised learning
Training steps
No
yes
Reconstructed output
Flatten
Design network structure/ Training parameters configuration
LSTM Dense
Supervised learning
Forward and back propagation with mini-batches
Optimal ?
N
y
Fine tunning
Model Testing
DAE-LSTM
LSTM Dense
Classification Results
End

Fig. 2. Process Flow of Our Proposed Scheme

B. DAE Training

In this stage, the DAE-LSTM model is trained, leveraging its feature extraction capabilities to extract meaningful features from the input data. The latent space represents a condensed and informative encoding of the input which plays a vital role in capturing relevant information for intrusion detection. The reconstructed output is flattened and sent to the next stage for further analysis. The flattened layer allows for connection to the LSTM dense layer and simplifies the data structure for further processing.

C. LSTM-Training

This stage takes in the clean reconstructed output as input for training. The LSTM dense is a layer within the DAE-LSTM, dedicated to carrying out the classification task. It is a fully connected layer that processes the information learned, and produces the final output, making decisions or predictions based on the learned features. This architecture is fine-tuned to achieve optimal performance.

D. Model Testing

During testing, the model is evaluated on unseen data to determine its performance. The DAE-LSTM is validated based on its ability to reconstruct noisy traffic signals while the LSTM dense is evaluated based on its classification ability.

E. Dataset Description

The ICS-flow dataset, derived from a simulated bottle-filling factory control system, encompasses raw network packets, flow records, and process variable logs. Processed with ICS-FlowGenerator, the dataset contains 45,719 flows, including normal operations (“0”) and distinct attacks (IP-Scan, Port-Scan Replay, DDoS, MitM denoted as (“1”)) for binary classification, representing real-world ICS vulnerabilities. The dataset captures network traffic utilizing the Modbus protocol during both normal and attack operations. Detailed information on dataset generation and extracted predictors is available in [15].

Before training, we analyzed the dataset to understand its learnability, gaining insights into high-dimensional patterns. The Andrews plot [16], in Fig. 3 revealed non-linearity and class overlap, guiding our model selection.

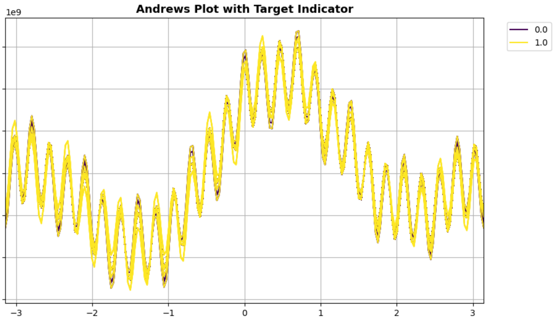

Fig. 3. Visualizing Non-Linearity in Dataset using Andrews plot

F. Experimental Set-up

The experiment was conducted in a Python environment using Tensorflow version 2.9.0. The operating system utilized was Windows 10. The hardware configuration includes; Intel(R) Core(TM) i5-7400CPU @ 3.00GHz processor, 8GB RAM, and a Tesla K80 GPU. Hyper-parameter tuning involved a manual process to identify optimal settings shown in Table I.

TABLE I
HYPERPARAMETER USED FOR PROPOSED SCHEME

S/n	Hyperparameters	Value(s)
1	Number of layers	6
2	Dropout rate	0.2
3	Activation function	Relu
4	Batch size	30
5	Optimizer	Adam
6	Learning rate	0.001
7	Epoch	50
8	Latent space size	19
9	Noise factor	0.1
10	LSTM units	257
11	Loss function	MSE / Binary cross-entropy

G. Performance Evaluation Metrics

The proposed model was validated based on its ability to reconstruct clean traffic signals at a decreased loss, along with its ability to classify ICS communication and events as normal behavior, or attack instances. While balanced data can greatly contribute to optimal model performance, we deliberately chose to work with an imbalanced dataset. This decision acknowledges that the class distribution naturally mirrors the occurrences in ICS, where anomalies are less frequent. Nevertheless, to handle this, we leveraged the performance metrics that effectively measure the model's performance on such data. Therefore in addition to accuracy (ACC), we used metrics such as; confusion matrix, Matthews correlation coefficient (MCC), recall (Rec), precision (Prec), F1-value and mean square error (MSE), as shown in equations 7, 8, 9, 10, 11, 12 and computing time (Comp.T) respectively.

$$\text{MSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{7}$$

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{9}$$

$$R = \frac{TP}{TP + FN} \tag{10}$$

$$P = \frac{TP}{TP + FP} \tag{11}$$

$$\Rightarrow F1value = \frac{2 * Precision * Recall}{Precision + Recall} \tag{12}$$

where, FN = False Negative, FP = False Positive, TN = True Negative, and TP = True Positive.

IV. Results and Analysis

The experimental results and a comparative analysis of the proposed DAE-LSTM with other approaches are discussed in this section. Table II, shows the results of the proposed model versus an LSTM model trained with similar parameter configurations.

TABLE II
COMPARATIVE ANALYSIS OF PROPOSED DAE-LSTM WITH LSTM UNDER SAME PARAMETER CONFIGURATION AND DATASET

Model	Acc (%)	Prec (%)	Rec (%)	F1-value (%)	MCC (%)	MSE	Comp.T (Sec)
LSTM	82.6	93.0	85.2	94.1	81.0	0.31	71
DAE-LSTM	**99.6**	**98.2**	**95.2**	**95.0**	**98.0**	**0.12**	**62**

The DAE-LSTM model outperforms LSTM across all metrics, with a high accuracy of 99.6%, 98.2% precision, and 95.1% recall. Also, the F1-value (which considers both precision and recall) of the proposed model, is slightly higher than that of the LSTM. Table II also highlights the minimal reconstruction loss of 0.12 recorded by the proposed model, outperforming the LSTM. We also took into consideration the model's computational time which is a vital factor when dealing with time-critical systems like ICS. We noticed that the LSTM model struggled with reconstructing noisy input and, hence took a longer time to train, compared to the DAE-LSTM model. This can be attributed to the intrinsic nature of the DAE, making it computationally efficient in handling noisy input.

Fig. 4 shows the DAE-LSTM reconstruction error, reaching a minimal value of 0.12. A lower MSE indicates better precision in reconstructing noisy input. With a train reconstruction loss of 0.31 and a validation reconstruction error of 0.12, the proposed model demonstrates effective learning and generalization capabilities. Indicating its potential to accurately reconstruct corrupted input network traffic signals in real-world scenarios.

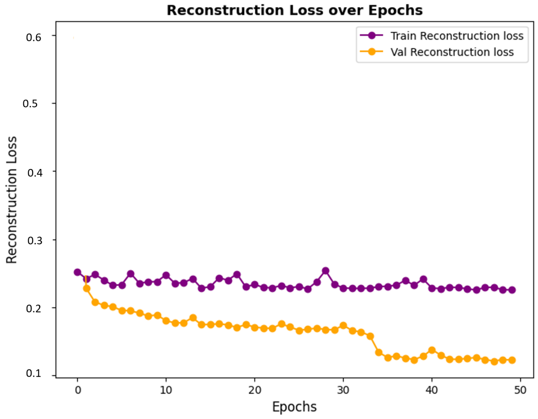

Fig. 4. Reconstruction Loss of proposed model

Fig. 5, shows the precision-recall curve of the proposed model, initiating at 100% for both precision and recall, maintaining high performance between 99% and 100%. This suggests stability, demonstrating accurate identification of positive instances while capturing a substantial proportion of other instances, emphasizing the model's effectiveness.

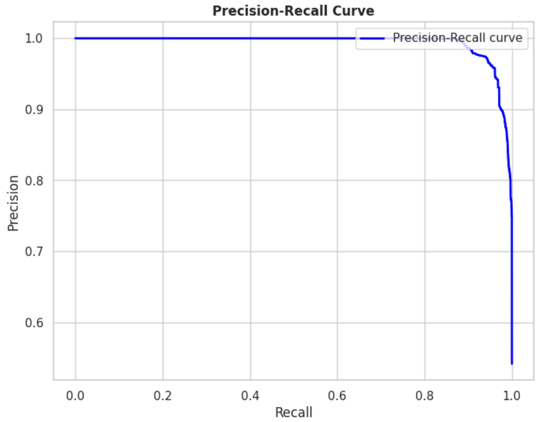

Fig. 5. The Precision-Recall Curve for the Proposed Model.

The confusion matrices in Fig. 6 and Fig. 7, show the number of rightly classified instances and misclassified predictions of the proposed framework and LSTM model, respectively. The diagonal elements from the top-left to bottom-right in the confusion matrix represent correct predictions (TP and TN), while the off-diagonal elements indicate incorrect predictions (FP and FN), respectively. The proposed DAE-LSTM has the least classification error compared with the LSTM Model.

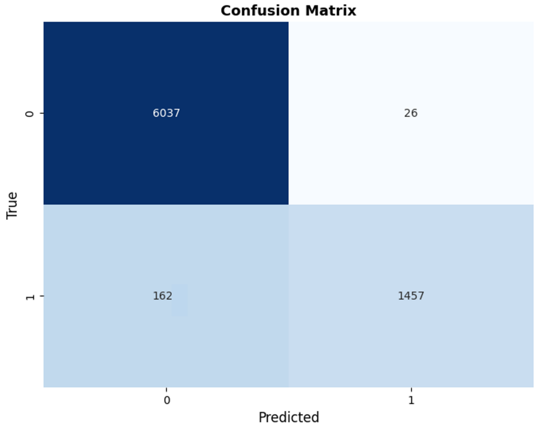

Fig. 6. Confusion Metrics of LSTM model

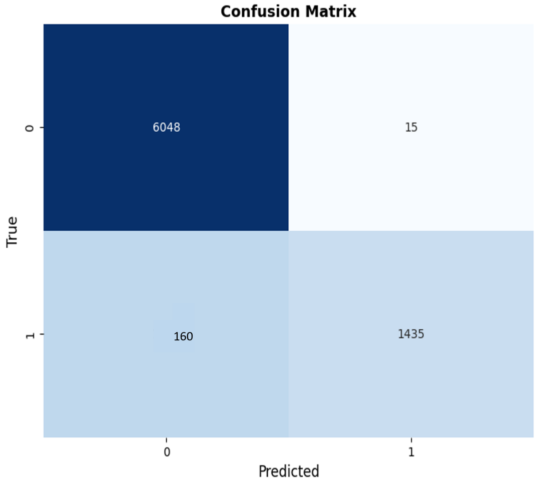

Fig. 7. Confusion Metrics of Proposed DAE-LSTM Model

Finally, to further validate the effectiveness of our proposed model, we compared it with the artificial neural network (ANN) proposed in [15], based on the same dataset. As shown in Table III, our proposed scheme achieved the best overall accuracy of 99.6% compared to their model, while performance on other metrics seems marginal. Although their proposed model demonstrated good overall performance on the metrics employed, they failed to consider the computational time which is an important metric when dealing with a time-critical system, such as ICS.

V. Conclusion

This study presents a security framework aimed at real-time attack detection in ICS. Our proposed approach integrates a regularization technique that considers the noisy nature of

TABLE III

Comparison of the Proposed Model with a DL-Based Method for Binary Classification Using the ICS-Flow Dataset

Model	**Scenario**	**Acc %**	**Pre %**	**Rec %**	**F1-value %**	**Comp. Time sec**
ANN [[15]	Normal	99.5	99.5	99.8	99.65	***
	Attack		99.2	98.0	99.65	
DAE-LSTM	Normal	**99.6**	**99.5**	**98.9**	**99.97**	**62 secs**
	Attack		**99.1**	**97.0**	**97.5**	

industrial processing, making it robust and effective in detecting and distinguishing threats targeted towards the network, from a normal network operation, amidst noise. Results obtained via experiments using the ICS-Flow dataset, show that our proposed framework is capable of reconstructing noisy input at a decreased error rate of 0.12, and a significant accuracy of 99.6% for binary classification task, compared to a different approach subjected to the same condition.

For our future work, first, we are working on enhancing our model explainability, by providing insights into the contribution of features in the decision-making process, and also establish a more robust comparison of our design with other existing autoencoder / LSTM methods from the literature. Secondly, we hope to integrate blockchain technology into our design. This entails using blockchain as a tamper-resistant ledger to store information related to network traffic and system logs. The aim is to establish a more robust, secure, and transparent record of network activities, providing an additional layer of trust and integrity for threat detection and mitigation in ICS networks.

Acknowledgment

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2024-2020-0-01612) supervised by the IITP (Institute for Information and Communications Technology Planning and Evaluation) and Priority Research Centers Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2018R1A6A1A03024003).

References

[1] A. Azmoodeh, A. Dehghantanha, and K.-K. R. Choo, "Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning," *IEEE Transactions on sustainable computing*, vol. 4, no. 1, pp. 88–95, 2018.

[2] R. Badarinath and V. V. Prabhu, "Advances in internet of things (iot) in manufacturing," in *Advances in Production Management Systems. The Path to Intelligent, Collaborative and Sustainable Manufacturing: IFIP WG 5.7 International Conference, APMS 2017, Hamburg, Germany, September 3-7, 2017, Proceedings, Part I.* Springer, 2017, pp. 111–118.

[3] D. Bhamare, M. Zolanvari, A. Erbad, R. Jain, K. Khan, and N. Meskin, "Cybersecurity for industrial control systems: A survey," *computers & security*, vol. 89, p. 101677, 2020.

[4] C. Feng, T. Li, and D. Chana, "Multi-level anomaly detection in industrial control systems via package signatures and lstm networks," in *2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN).* IEEE, 2017, pp. 261–272.

[5] D. U. Case, "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, vol. 388, no. 1-29, p. 3, 2016.

[6] M. Niedermaier, F. Fischer, and A. von Bodisco, "Propfuzz—an it-security fuzzing framework for proprietary ics protocols," in *2017 International conference on applied electronics (AE).* IEEE, 2017, pp. 1–4.

[7] Z. Drias, A. Serhrouchni, and O. Vogel, "Taxonomy of attacks on industrial control protocols," in *2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS).* IEEE, 2015, pp. 1–6.

[8] J. Kim, J. Shin, K.-W. Park, and J. T. Seo, "Improving method of anomaly detection performance for industrial iot environment." *Computers, Materials & Continua*, vol. 72, no. 3, 2022.

[9] S. Mokhtari, A. Abbaspour, K. K. Yen, and A. Sargolzaei, "A machine learning approach for anomaly detection in industrial control systems based on measurement data," *Electronics*, vol. 10, no. 4, p. 407, 2021.

[10] L. A. C. Ahakonye, C. I. Nwakanma, J.-M. Lee, and D.-S. Kim, "Scada intrusion detection scheme exploiting the fusion of modified decision tree and chi-square feature selection," *Internet of Things*, vol. 21, p. 100676, 2023.

[11] I. A. Khan, M. Keshk, D. Pi, N. Khan, Y. Hussain, and H. Soliman, "Enhancing iiot networks protection: A robust security model for attack detection in internet industrial control systems," *Ad Hoc Networks*, vol. 134, p. 102930, 2022.

[12] K. Han, Y. Wang, C. Zhang, C. Li, and C. Xu, "Autoencoder inspired unsupervised feature selection," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP).* IEEE, 2018, pp. 2941–2945.

[13] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, 2014, pp. 4–11.

[14] X. Zhang, Y. H. Wu, D.-P. Covei, X. Hao *et al.*, "Complex boundary value problems of nonlinear differential equations: Theory, computational methods, and applications," in *Abstract and Applied Analysis*, vol. 2013. Hindawi.

[15] A. Dehlaghi-Ghadim, M. H. Moghadam, A. Balador, and H. Hansson, "Anomaly detection dataset for industrial control systems," *arXiv preprint arXiv:2305.09678*, 2023.

[16] V. Grinshpun, "Application of andrew's plots to visualization of multidimensional data," *International Journal of Environmental and Science Education*, vol. 11, no. 17, pp. 10 539–10 551, 2016.