

Intelligent Security Hardening of SCADA Systems Using Machine Learning Algorithms [†]

Sinan Atıcı ¹  and Gurkan Tuna ^{2,*} ¹ SecHard Information Technologies, Istanbul 34750, Türkiye; sinan.atici@sechard.com² Department of Computer Technologies, Vocational College of Technical Sciences, Trakya University, Edirne 22100, Türkiye

* Correspondence: gurkan.tuna@trakya.edu.tr

[†] Presented at the 3rd International Electronic Conference on Machines and Applications, 12–14 May 2026; Available online: <https://sciforum.net/event/IECMA2026>.

Abstract

Supervisory Control and Data Acquisition (SCADA) systems play a key role in various industrial processes. Due to their key role in critical infrastructures, in recent years, they have become the target of cyber attackers. In this research, we propose an intelligent security hardening approach for SCADA systems using machine learning algorithms. The proposed approach relies on the collection and analysis of network traffic data from SCADA systems, followed by the application of machine learning algorithms to detect and respond to cyber threats. The proposed approach can improve the security of SCADA systems and reduce the risk of downtime and financial losses due to cyber attacks. It is a more cost-effective security solution compared to traditional security measures.

Keywords: SCADA systems; security threats; hardening; machine learning algorithms

1. Introduction

Supervisory Control and Data Acquisition (SCADA) systems can be thought of as the backbone of current industrial and critical infrastructure systems and provide real-time monitoring and control of process operations in different fields [1]. The principal objective of these systems is to guarantee the efficient operation and reliability of their process operations [2]. For this reason, these systems comprise different kinds of devices, such as sensors, actuators, programmable logic controllers, and human–machine interfaces, together with software applications. Previously, these systems functioned as standalone systems using proprietary communication networks [3]. Consequently, they were considered safe platforms.

Recently, owing to the importance of SCADA systems in national critical infrastructures and their compatibility with Internet Protocol (IP)-based networks and the Industrial Internet of Things (IIoT) [4], SCADA systems have been vulnerable to different kinds of cyber attacks [5]. While the integration of IT and OT has resulted in improving the functionality of these systems, on the other hand, it has created some new cybersecurity vulnerabilities [6].

Although conventional security devices and software solutions such as firewalls, signature-based intrusion detection systems, and access control systems have been employed to protect the SCADA system, these systems alone cannot defend the SCADA systems from advanced and sophisticated cybersecurity attacks, including zero-day attacks and advanced persistent threats [7]. For example, zero-day attacks take advantage



Academic Editor: Antonio J. Marques Cardoso

Published: 9 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

of vulnerabilities that have not yet been discovered, while advanced persistent threats constantly change.

Machine learning (ML) models could be used to strengthen SCADA security. They can use normal system behavior patterns and find any deviations from them, which would indicate cyber intrusion or other anomalies [1]. Such functions are crucial for SCADA systems, since SCADA systems produce a massive amount of data in real time. However, the implementation of ML models in these systems raises various difficulties. These difficulties relate to the data availability, interpretability, computational restrictions, and susceptibility of ML models to adversarial attacks [5]. Moreover, SCADA systems have real-time constraints, which reduce the applicability of these models with high computation complexity. Adaptive and proactive measures of defense represent what is referred to as intelligent security hardening, which could be implemented in SCADA systems through these models.

As discussed above, although some security hardening solutions have been proposed to secure SCADA systems, conventional security measures such as firewalls, intrusion detection systems, and access controls are not enough to provide adequate protection against modern cyber threats. Therefore, in this paper, we propose an intelligent security hardening approach for SCADA systems using ML models and evaluate its effectiveness.

2. Related Work

SCADA systems have unique cybersecurity challenges [8]. These challenges arise from several factors. First, compared to IT systems, they have different architecture and operational requirements. Second, most SCADA systems still rely on legacy devices and components [9]. Finally, many of them continue to rely on outdated software and operate using legacy communication protocols. Due to these problems, the growing interconnectivity of SCADA systems has exposed them to various cyber threats, including malware, denial-of-service attacks, insider threats, and advanced persistent threats. These threats exploit vulnerabilities in the communication protocols, network configurations, and system components of SCADA systems. Consequently, there is a need for advanced security solutions that can adapt to evolving, sophisticated cyber threats.

ML models are important for intrusion detection in SCADA systems because they offer improved detection capabilities to these systems compared to traditional cybersecurity mechanisms [9]. A review of the literature shows that supervised ML models, such as support vector machines (SVM), decision trees, and random forests, have been extensively used for classifying network traffic and identifying malicious activities [8]. These models are trained on labeled datasets and can achieve high detection accuracy under controlled conditions. The use of unsupervised ML models has also been investigated because they can detect unknown cyber attacks without requiring labeled data. These models are particularly useful if new attack patterns emerge frequently [1]. As well as supervised and unsupervised ML models, deep learning models, such as neural networks and recurrent neural networks, have been employed to enhance the intrusion detection capabilities of intrusion detection mechanisms.

Deep learning models can capture complex temporal and spatial patterns that are difficult to identify using conventional supervised and unsupervised ML models [10]. However, deep learning models have high computational complexity and lead to real-time processing constraints [10].

As anomaly detection allows identifying deviations from normal system behavior, it has an important role in SCADA security. To enable anomaly detection in SCADA systems, ML models are used. ML models can analyze time-series data generated by sensors and SCADA control systems to detect anomalies. Anomalies, particularly, could indicate

cyber attacks or system failures [1]. Different from signature-based IDS, ML models can detect zero-day attacks. By learning historical attack patterns, they can anticipate potential threats. Therefore, they are generally viewed as predictive security solutions that allow for early detection and mitigation of cyber threats. Although they can enhance SCADA system resilience to potential cyber threats, the integration of ML-based intrusion detection mechanisms into SCADA systems is associated with several challenges. First, as access to real-world SCADA data is often restricted due to security and privacy concerns, high-quality SCADA datasets are not available [1]. Second, SCADA systems operate in time-critical environments, and any delay in anomaly detection or response can have significant consequences. Therefore, ML models must be optimized for efficiency and low latency. Third, model interpretability is a critical concern because operators require transparent and explainable systems. Finally, ML models are vulnerable to adversarial attacks. In adversarial attacks, malicious actors manipulate input data to evade detection [8].

Intelligent security hardening in SCADA systems relies on continuous monitoring, adaptive intrusion detection, and automated response mechanisms. Such an adaptive defense strategy can enhance the resilience of SCADA systems against various cyber threats significantly [1]. Recently, combining edge and fog computing with ML models has been proposed. This approach enables distributed and scalable SCADA intrusion detection solutions. Fog and edge computing can help reduce latency and improve responsiveness in such solutions. This makes fog and edge computing suitable for real-time SCADA environments.

3. Methodology

This study proposes an intelligent security hardening framework for SCADA systems based on ML. As shown in Figure 1, the proposed framework relies on a four-stage pipeline: (1) data acquisition and preprocessing, (2) feature extraction and engineering, (3) multi-algorithm model training and ensemble inference, and (4) automated threat response. Overall, the framework emphasizes efficiency, minimal false positives, and operability within resource-constrained operational technology environments.

3.1. Data Acquisition and Preprocessing

In the proposed framework, data packets from the nodes in the SCADA environment are continuously captured, and this process involves four major data sources: raw data packet captures, industrial protocol traffic (MODBUS/DNP3), sensor readings and control logs, and system-level event data. In order to ensure the reproducibility of results obtained and to compare them to other related work, models are developed and evaluated using two public benchmark datasets for industrial control systems: the BATADAL dataset, which comprises cyber attack scenarios against a water distribution control system, and the Secure Water Treatment (SWaT) dataset from the iTrust Centre, collected over six days of traffic.

To obtain a fair and replicable evaluation, the datasets BATADAL and SWaT were split into two subsets, training and testing, by applying stratification and an 80/20 ratio. The first subset contained 80% of the dataset and was used for developing models and setting hyperparameters. The second subset (20%) was used only for the final evaluation of the models and did not take part in the process of training. Such splitting helps prevent information leakage and obtain unbiased estimates of the generalization performance of models.

Within the training subset, a stratified 5-fold cross-validation approach was applied for selecting models and optimizing their hyperparameters. In each iteration of cross-validation, 80% of the data were used for training and 20% for validating the models. Five iterations were run in such a way that each example was included in the validation set

once and in the training set four times. Average performance estimates based on the five iterations of cross-validation help to decrease the variance that occurs due to single splitting into train/validation sets. The final configuration of each model was decided through the averaging of the cross-validation F1-score. After selecting the models, the selected models were trained again using the whole training subset and tested on an independent test set. The precision, recall, F1-score, and latency of detection were the key performance measures.

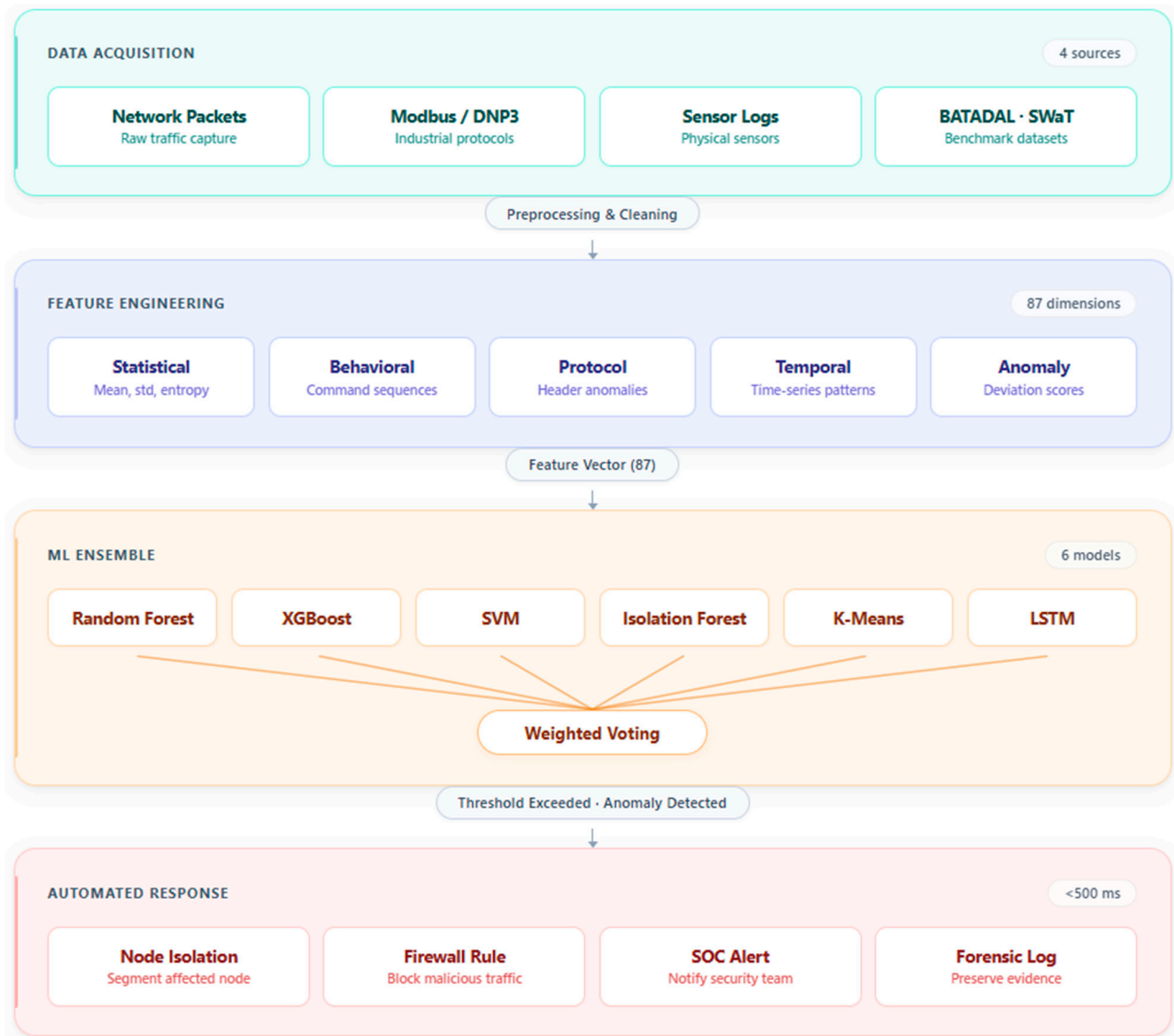


Figure 1. Proposed four-stage ML-based security hardening pipeline for SCADA systems.

Data preparation involves the application of a standard pre-processing chain before the feeding of the model. Missing values are filled with forward-filling based on the nature of SCADA sensor time series data. Protocol types and devices' IDs are one-hot encoded. All continuous variables are scaled to the (0, 1) scale using min-max normalization to ensure that large-magnitude variables do not dominate the smaller ones. Network traffic was divided into 30 s intervals with 50% overlap to ensure continuity. The consecutive intervals were later fed to the LSTM model in order to consider the long-term dependency.

3.2. Feature Engineering

Five types of features are derived from the preprocessed data. Statistical features represent distribution information of network flows in terms of mean and variance of

packet size, inter-arrival time, and flow duration. Behavioral features model device-level interaction behaviors like command frequency histograms and communication graph centralities. Protocol-related features are extracted from the fields of industrial protocols, such as the distribution of Modbus function codes and the frequency of DNP3 object types, which are especially useful in detecting protocol-level manipulation attacks. Temporal features exploit the time-series nature of SCADA data via rolling window statistics, seasonal decomposition residuals, and auto-correlation coefficients. Anomaly features indicate deviation through z-score computation of key metrics against the rolling window baseline for 24 h and rare events. Each window segment is represented by an 87-dimensional feature vector. The 87-dimensional feature set was obtained after preliminary feature selection experiments and based on commonly used SCADA traffic descriptors reported in the literature.

3.3. ML Models and Ensemble Inference

A total of six ML models are trained and integrated in an ensemble inference engine, offering a layered approach towards threat protection. In the case of supervised classification of known attacks, a random forest classifier and XGBoost gradient boosting model are used, with the former being preferred because of its ability to produce interpretable results in a safety-critical environment and the latter offering more precision by using ensemble boosting [11,12]. The SVM model uses a radial basis function kernel to separate anomalies from normal traffic using binary classification. In the event of zero-day attacks for which no training samples exist, the isolation forest uses unsupervised anomaly scoring using path length deviation, while k-means clustering determines behavior baseline deviations to be measured. Lastly, a two-layered LSTM network is used to process sequential windows in order to detect multi-step attacks that would be impossible to detect using windowing alone.

These six models were chosen based on their different learning methods. Random Forest, XGBoost, and SVM were used in supervised learning tasks where attack patterns are known. Isolation Forest and K-means were used in cases when there are no labels or very few labeled attacks available. LSTM was used to account for temporal dependencies between observation windows. A two-layer LSTM architecture was employed to model temporal dependencies across observation windows. Dropout regularization (0.3) was applied to reduce overfitting.

In the F1-weighted voting method, higher importance is assigned to models that perform better overall on the validation dataset. This is because it is known that F1-scores do not necessarily represent complementarity per attack type; it is considered a future direction to use a more detailed weight assignment based on precision and recall per class. The hyperparameters used by each of the models are given in Table 1.

Table 1. Hyperparameter configurations of the six ensemble models.

Model	Key Hyperparameters
Random Forest	n_estimators = 200; max_depth = 20; min_samples_leaf = 2
XGBoost	learning_rate = 0.05; max_depth = 6; n_estimators = 300; subsample = 0.8
SVM	kernel = RBF; C = 10; gamma = 'scale'
Isolation Forest	n_estimators = 100; contamination = 0.05
K-Means	k = 3 (elbow method); clusters: normal, marginal, anomalous
LSTM (2-layer)	hidden = [128, 64]; dropout = 0.3; seq_len = 10; optimizer = Adam (lr = 0.001); epochs = 50 (early stopping)

At the inference phase, the output scores from all six ML models are combined into a composite risk score by means of a weighted voting approach in which the weights of each individual model are calculated based on its F1-score over the held-out validation set. The traffic window is considered to be malicious if the risk score exceeds an adaptive threshold value, which is updated regularly based on the sliding validation buffer due to the concept drift in the SCADA system. Once the system detects that the adaptive threshold has been surpassed, the automated response system triggers a priority list of actions: traffic isolation of the source node, insertion of new rules into the network firewall, sending out alerts to the security operations center, and capturing packets for forensics. The framework was designed with a target end-to-end latency below 500 ms.

Concept drift was accounted for by updating the decision threshold using a sliding validation window and exponential weighted moving average (EWMA) smoothing. These three parameters need to be empirically set and thus are implementation-specific.

Resistance of ML models to attacks by adversaries is still an important issue in the application of ML to SCADA security [13]. Our proposed system includes various types of supervised, unsupervised, and deep learning models that may help in the detection of adversaries by virtue of their complementary nature. Moreover, protocol-based features of Modbus and DNP3 communications were incorporated in order to characterize traffic on industrial networks. However, no adversarial robustness tests were carried out in this research study. Hence, any conclusions about adversarial resistance of the proposed framework cannot be made at this time.

4. Experimental Results

4.1. Model Configuration

For reproducibility purposes, the configurations used for the hyperparameters of each of the six models are presented in Table 1. In Random Forest, there are 200 estimators with a max depth of 20 and a minimum number of 2 samples per leaf to find a balance between predictive capability and overfitting, taking into account that the BATADAL and SWaT datasets are medium-sized. XGBoost makes use of a learning rate of 0.05, a maximum tree depth of 6, 300 boosting rounds, and a subsample ratio of 0.8. SVM employs an RBF kernel with regularization parameters of $C = 10$ and $\gamma = \text{'scale'}$ according to the state-of-the-art literature on normalized feature space. Isolation Forest has 100 estimators with a contamination parameter of 0.05. In K-means, there are $k = 3$ clusters based on the results of elbow-method analysis corresponding to normal operation, marginal deviation, and anomalies. Two-layer LSTM has hidden dimensions of 128 and 64 in the first and second layers, respectively; a dropout rate of 0.3; a sequence length of 10 windows; and the Adam optimizer at a learning rate of 0.001, trained for up to 50 epochs with early stopping.

4.2. Individual Model Evaluation on BATADAL and SWaT

Each of the six individual models was trained and evaluated separately against the BATADAL and SWaT datasets with the use of the pre-processing and feature engineering pipeline defined in Sections 3.1 and 3.2. Precision, recall, and F1-score were used as evaluation metrics. For the BATADAL dataset, the Random Forest model achieved an F1-score of 0.91, the XGBoost model achieved 0.93, the SVM achieved 0.87, Isolation Forest achieved 0.74, and K-means achieved 0.68, while the LSTM achieved 0.89. For SWaT, the Random Forest model achieved an F1-score of 0.88, the XGBoost achieved 0.90, SVM achieved 0.85, Isolation Forest achieved 0.71, and K-means achieved 0.64, while the LSTM achieved 0.86. The weighted voting ensemble of all six models achieved an F1-score of 0.94 on BATADAL and 0.92 on SWaT, outperforming all of its constituent components. This gives empirical ground for the proposed architecture and is consistent with the baseline

results of the benchmark literature [14,15]. A comprehensive hyperparameter optimization, ablation studies of window parameters, and generalization analysis on other datasets will be the main focus of the future work.

5. Conclusions

Over the past decade, the complexity and interconnectivity of SCADA systems have increased. This has made them vulnerable to various types of cyber threats. When those potential threats are considered, although conventional security mechanisms are essential, they alone are not sufficient to address the dynamic and evolving nature of modern cyber attacks. As it enables adaptive, intelligent, and data-driven approaches, recently ML has emerged as a powerful tool for enhancing SCADA security.

ML-based techniques offer advantages over conventional security mechanisms and improve the intrusion detection, anomaly detection, and predictive security capabilities of the traditional ones. However, addressing several challenges is essential for ensuring their effective implementation, including real-time processing constraints, model interpretability, data scarcity, and vulnerability to adversarial attacks.

In this paper, we proposed an intelligent security hardening approach for SCADA systems using ML models and evaluated its effectiveness. As our experimental results show, the proposed approach can improve the security of SCADA systems and reduce the risk of downtime and financial losses due to cyber attacks. It is a more cost-effective security solution compared to conventional security measures.

Future research should focus on developing explainable ML models, creating standardized datasets, and exploring hybrid approaches that combine traditional and ML-based security mechanisms, as advancing intelligent security solutions is essential for ensuring the safety, reliability, and resilience of critical infrastructure systems.

Author Contributions: Conceptualization, S.A. and G.T.; methodology, S.A. and G.T.; software, S.A.; investigation, S.A. and G.T.; writing—original draft preparation, S.A. and G.T.; writing—review and editing, G.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This study uses publicly available BATADAL and SWaT datasets.

Conflicts of Interest: The authors declare no conflicts of interest. SecHard Information Technologies has no conflict of interest.

References

1. Teixeira, M.A.; Salman, T.; Zolanvari, M.; Jain, R.; Meskin, N.; Samaka, M. SCADA system testbed for cybersecurity research using machine learning approach. *Future Internet* **2018**, *10*, 76. [\[CrossRef\]](#)
2. Yadav, G.; Paul, K. Architecture and security of SCADA systems: A review. *Int. J. Crit. Infrastruct. Prot.* **2021**, *34*, 100433. [\[CrossRef\]](#)
3. Wali, A.; Alshehry, F. A Survey of Security Challenges in Cloud-Based SCADA Systems. *Computers* **2024**, *13*, 97. [\[CrossRef\]](#)
4. He, W.; Baig, M.J.A.; Iqbal, M.T. An Internet of Things—Supervisory Control and Data Acquisition (IoT-SCADA) Architecture for Photovoltaic System Monitoring, Control, and Inspection in Real Time. *Electronics* **2025**, *14*, 42. [\[CrossRef\]](#)
5. Ferrag, M.A.; Babaghayou, M.; Yazici, M.A. Cyber security for fog-based smart grid SCADA systems: Solutions and challenges. *J. Inf. Secur. Appl.* **2020**, *52*, 102500. [\[CrossRef\]](#)
6. Rodriguez-Casavilca, H.M.; Mauricio, D.; Villanueva, J.M.M. Evolution of Artificial Intelligence-Based OT Cybersecurity Models in Energy Infrastructures: Services, Technical Means, Facilities and Algorithms. *Energies* **2025**, *18*, 5163. [\[CrossRef\]](#)
7. Yaacoub, J.A.; Salman, O.; Noura, H.N.; Kaaniche, N.; Chehab, A.; Malli, M. Cyber-physical systems security: Limitations, issues and future trends. *Microprocess. Microsyst.* **2020**, *77*, 103201. [\[CrossRef\]](#) [\[PubMed\]](#)

8. Khan, A.A.Z.; Serpen, G. Intrusion detection and identification system design for industrial SCADA networks. *Int. J. Saf. Secur. Eng.* **2022**, *12*, 215–224. [[CrossRef](#)]
9. Nankya, M.; Chataut, R.; Akl, R. Securing Industrial Control Systems: Components, Cyber Threats, and Machine Learning-Driven Defense Strategies. *Sensors* **2023**, *23*, 8840. [[CrossRef](#)] [[PubMed](#)]
10. Khadidos, A.O.; Khadidos, A.O.; Selvarajan, S.; Al-Shehari, T.; A Alsadhan, N.; Singh, S. CyberSentry: Enhancing SCADA security through advanced deep learning and optimization strategies. *Int. J. Crit. Infrastruct. Prot.* **2025**, *50*, 100782. [[CrossRef](#)]
11. Imani, M.; Beikmohammadi, A.; Arabnia, H.R. Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels. *Technologies* **2025**, *13*, 88. [[CrossRef](#)]
12. Wu, Y.; Chow, K.-H.; Wei, W.; Liu, L. Exploring Model Learning Heterogeneity for Boosting Ensemble Robustness. In *Proceedings of the 2023 IEEE International Conference on Data Mining (ICDM), Shanghai, China, 1–4 December 2023*; IEEE: New York, NY, USA, 2023; pp. 648–657.
13. Anthi, E.; Williams, L.; Rhode, M.; Burnap, P.; Wedgbury, A. Adversarial attacks on machine learning cybersecurity defences in industrial control systems. *J. Inf. Secur. Appl.* **2021**, *58*, 102717. [[CrossRef](#)]
14. Taormina, R.; Galelli, S.; Tippenhauer, N.O.; Salomons, E.; Ostfeld, A.; Eliades, D.G.; Aghashahi, M.; Sundararajan, R.; Pourahmadi, M.; Banks, M.K.; et al. Battle of the Attack Detection Algorithms: Disclosing Cyber Attacks on Water Distribution Networks. *J. Water Resour. Plan. Manag.* **2018**, *144*, 04018048. [[CrossRef](#)]
15. Mathur, A.P.; Tippenhauer, N.O. SWaT: A Water Treatment Testbed for Research and Training on ICS Security. In *Proceedings of the 2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater), Vienna, Austria, 11 April 2016*; IEEE: New York, NY, USA, 2016; pp. 31–36.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.