Challenges in Machine Learning based approaches for Real-Time Anomaly Detection in Industrial Control Systems

Chuadhry Mujeeb Ahmed*, Gauthama Raman M R†, Aditya Mathur‡
iTrust, Center for Research in Cyber Security
Singapore University of Technology and Design
Singapore
*chuadhry@mymail.sutd.edu.sg, †gauthama_mani@sutd.edu.sg, ‡aditya_mathur@sutd.edu.sg

***Abstract*—Data-centric approaches are becoming increasingly common in the creation of defense mechanisms for critical infrastructure such as the electric power grid and water treatment plants. Such approaches often use well-known methods from machine learning and system identification, i.e., the Multi-Layer Perceptron, Convolutional Neural Network, and Deep Auto Encoders to create process anomaly detectors. Such detectors are then evaluated using data generated from an operational plant or a simulator; rarely is the assessment conducted in real time on a live plant. Regardless of the method to create an anomaly detector, and the data used for performance evaluation, there remain significant challenges that ought to be overcome before such detectors can be deployed with confidence in city-scale plants or large electric power grids. This position paper enumerates such challenges that the authors have faced when creating data-centric anomaly detectors and using them in a live plant.**

***Index Terms*—Anomaly detection; Critical Infrastructure; Industrial Control Systems; Machine Learning; System Identification; Water Treatment Plant.**

I. Introduction

Industrial Control Systems (ICS) are found in modern critical infrastructure (CI) such as the electric power grid and water treatment plants. The primary role of an ICS is to control the underlying process in a CI. Such control is effected through the use of computing and communication elements such as Programmable Logic Controllers (PLCs) and Supervisory Control and Data Acquisition systems (SCADA), and communications networks. The PLCs receive data from sensors, compute control actions, and send these over to the actuators for effecting control over the process. The SCADA workstations are used to exert high level control over the PLCs, and the process, and provide a view into the current process state. Each of these computing elements is vulnerable to cyber attacks as evident from several widely reported successful attempts such as those reported in, [1]–[3]. Such attacks have demonstrated that while air-gapping a system might be a means to consider securing a CI, it does not guarantee keeping attackers from gaining access to the ICS.

Successful attacks on CI have led to a surge in the development of defense mechanisms to prevent, contain, and react to cyber attacks. One such defense mechanism is the anomaly detector that aims at raising an alert when the controlled process in a CI moves from its normal to an unexpected, i.e. *anomalous*, state. Approaches used in the design of such detectors fall into two broad categories: design-centric [4] and data-centric [5]. The focus of this position paper is on the data-centric approaches that rely on well-known methods for model creation such as those found in the system identification [6] and machine learning literature.

The use of machine learning to create anomaly detectors becomes attractive with increasing availability of data and advanced computational resources. However, our attempts to create anomaly detectors, and test them in a live water treatment plant, point to several challenges that must be overcome before such detectors can be deployed with confidence in a live plant. It is these challenges that are described in the remainder of this paper with the hope that other researchers will come forward and propose practical solutions to overcome these challenges.

Organization: The remainder of this paper is organized as follows. Section II is a brief introduction to a live water treatment plant used extensively by the authors for testing anomaly detectors derived using process data. Terminology related to anomaly detectors is explained in Section III. Challenges in the design of anomaly detectors using plant data are enumerated and explained in Section IV. Research directions aimed at the development of methods to overcome the challenges are summarized in SectionV.

II. SWaT: A live water treatment plant

The Secure Water Treatment (SWaT) plant is a testbed at the Singapore University of Technology and Design [7]. SWaT has been used extensively by researchers to test defense mechanisms for CI [8]. A brief introduction is provided in the following to aid in understanding the challenges described in this work.

SWaT is a scaled-down version of a modern water treatment process. It produces 5 gallons/minute of water purified first using ultrafiltration followed by reverse osmosis. The ICS in SWaT is a distributed control system consisting of six stages.

Each stage is labeled as P_n, where n denotes the nth stage. Each stage is equipped with a set of sensors and actuators. Sensors include those to measure water quantity, such as, level in a tank, flow, and pressure, and those to measure water quality parameters such as pH, oxidation reduction potential, and conductivity. Motorized valves and electric pumps serve as actuators.

Stage 1 processes raw water for treatment. Chemical dosing takes place in stage 2 to treat the water depending on the measurements from the water quality sensors. Ultrafiltration occurs in stage 3. In stage 4 any free chlorine is removed from water before it is passed to the reverse osmosis units in stage 5. Stage 6 holds the treated water for distribution and cleaning the ultrafiltration unit through a backwash process. Data from the sensors and actuators is communicated to the PLCs through a level 0 network. PLCs communicate with each other over a level 1 network.

III. Anomalies, detection, and development stages

A physical process in a CI is controlled to stay within its design limits. Thus, bounds are placed on each state variable in the process while the controller ensures that there is no drift beyond these. For example, water level in a tank must never exceed a high mark or fall below a low mark. In addition, each state variable evolves over time in accordance with process design. For example, the ultrafiltration unit must be cleaned at least every 30 minutes thus ensuring that the pressure drop across the unit remains within safety limits.

Any violation in the bounds or evolution of one or more state variables is considered process anomaly. Assuming correct controller design, such anomalies could arise due to faults in the physical components in the plant or due to cyber attacks. It is such anomalies that ought to be detected rapidly, and the operators alerted, before any remedial actions are initiated. The key objective of an anomaly detector is to ensure that any process anomaly is detect preferably as soon as it occurs and the plant operator notified.

An anomaly detector can be considered as a black box that receives data in real time from an operational plant, such as SWaT. Thus, the detector is continuously monitoring the state of the plant and comparing it with the predicted state; the prediction being carried out using the model built. The model itself is built using data from an operational plant where the detector is intended to be deployed. Once the model is built, it is tested to ensure that expected anomalies are detected and alerts generated in a timely manner. The tested model is then deployed in the plant.

The model becomes increasingly useless if it generates false positives that annoy the operators and waste their time in debugging the process that is otherwise operating normally. Thus, an anomaly detector must have ultra-high detection rate as well as ultra-low rate of false alarms. There is no widely accepted numbers associated with such rates though we believe that a detection rate of at least 99%, and a false alarm rate of less than one false alarm in 6-months, is needed for an anomaly detector deployed for real time monitoring of a city-scale plant or power grid. Thus, as the plant components degrade, or the plant is upgraded, the anomaly detector must adapt itself to the new reality.

IV. Challenges in the design of anomaly detectors

The development of an anomaly detector goes through several phases. The process begins with model creation, validation, and testing, followed by deployment and tuning, and lastly operation and retraining. Figure 1 shows the activities during these three phases. During the model creation phase, one needs to decide on the model to be used. There exist a variety of methods one could use to create a model that would eventually serve as an anomaly detector. Often used methods include those based on deep learning techniques, such as MLP, autoencoder (AE), CNN, Generative Adversarial Network (GAN), Recurrent Neural Network (RNN), Long short-term memory (LSTM)) as well as shallow learning models (Support vector machine (SVM), Decision tree (DT), Random forest (RF)) [10]–[12].

System identification [6] is another well known method often used by control engineers to build the state space model of a system using state observations. Once the model is created, and tested, it needs to be deployed in an operational plant. Doing so requires overcoming logistical challenges. Once the model is deployed and is operational, it's parameters will likely need re-tuning, and possibly retraining, as the plant being defended ages or is upgraded with new equipment and modified control policies.

Given the above phases, challenges described here are classified into the following categories: model creation, model deployment, and model retraining.

A. Challenges: Model creation

Challenges faced during the creation of an anomaly detector are described next.

Challenge 1: Supervised vs unsupervised Learning: Recently, there have been studies where supervised machine learning is used for attack detection [13]–[15]. Although these models possess high detection rate and generate few false alarms for known attacks, they fail to detect the unknown or new attacks due to the lack of signatures. Further, upgrading to the signature database at regular intervals seems to be a better solution though the generation of signatures for process anomalies in case of multivariate operational ICS environment is highly complex and impractical task.

Position 1: Supervised learning is not suitable for detecting zero day vulnerabilities in ICS.

Recent studies have used unsupervised or semi-supervised machine learning algorithms to detect attacks in an ICS [5], [16]–[18]. In particular, data was obtained from Secure Water Treatment (SWaT) testbed [19]. Generally, unsupervised learning models are designed based on the normal operation of the plant's behavior wherein any observation that

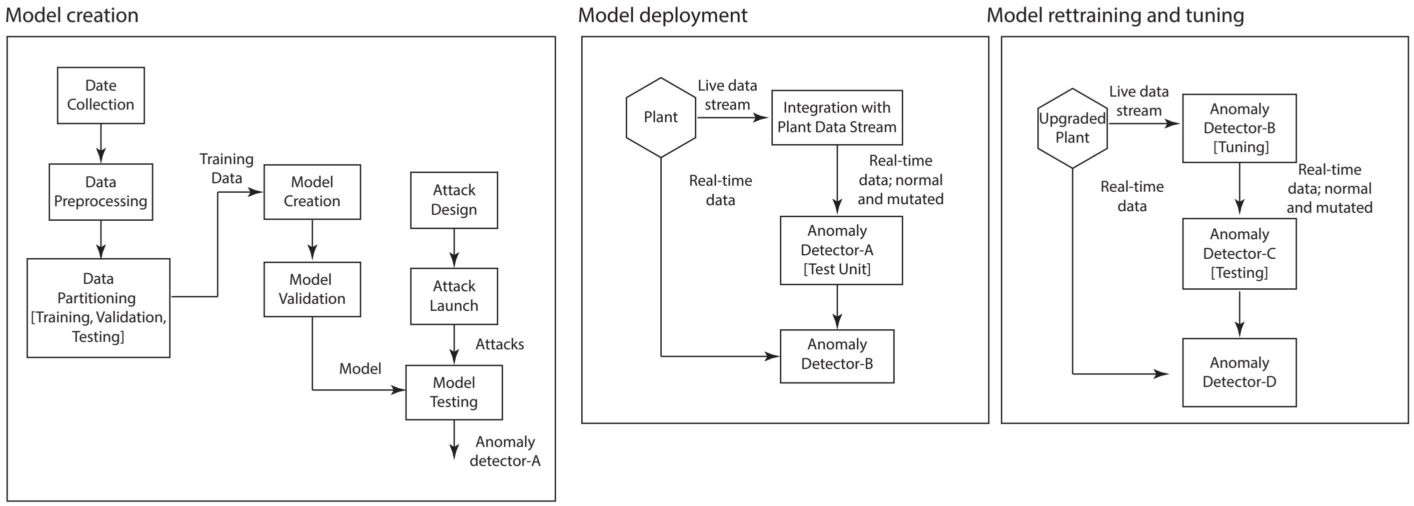

Fig. 1. Three phases in the development of an anomaly detector: creation, deployment, and retraining/tuning. Real-time data from the host plant needs to be mutated to create attacks for testing the anomaly detector before enabling it operationally. This step is needed for in-plant testing of the detector [9]. A, B, C and D denote the successively enhanced detectors created in each phase.

deviates from the "normal" is termed as an anomaly. In [5] the authors compared models derived using both supervised and unsupervised learning. In this study it was observed that models created using unsupervised learning perform better in attack detection though, due to sensitivity to noise in the data, they lead to a higher rate of false alarms than those derived using supervised learning.

Position 2: Unsupervised learning can detect unknown attacks but can also increase the number of false alarms.

In recent years, the design of anomaly detector for ICS is treated as an "one class classification problem" and several unsupervised learning methods are effectively employed [20]. Unsupervised learning approaches construct a baseline for the normal behaviour through feature learning and monitor whether the current behaviour is within the specified range or not. Although these techniques are able to detect zero day vulnerabilities, they generate high false alarms due to the existence of several hyperparameters and multivariate nature of ICS data. In [12] the authors have investigated the performance of several unsupervised neural network models for anomaly detection in SWaT testbed and proposed various statistical anomaly scoring techniques to achieve minimal false alarms.

Challenge 2: Model localization: An ICS in large systems is mostly a complex distributed control system. For example, a water treatment process consists of several stages and sub-processes. These separate physical processes might be connected logically and physically. An important consideration here is whether to create a machine learning model for the entire process or one each for different stages.

Position 3: Considering the distributed model versus a model for the whole system, or having a cluster of models, is an important design consideration that can influence detector performance.

Recently, iTrust researchers carried out an investigation on the significance of design knowledge in the data centric approaches for anomaly detection in SWaT testbed. In this work, three different variants of deep autoencoder (DAE) were designed and evaluated. These are: (i) DAE-$\boldsymbol{C_{AD}}$ - six AE models monitoring each stage independently, (ii) DAE-$\boldsymbol{C_{AD}}$ - three AE models monitoring the stage 1-2-3, stage 3-4-5, and stage 5-6-1 independently and (iii) DAE-$\boldsymbol{O_{AD}}$ - one AE model monitoring the entire SWaT operation. The creation of each DAE uses uses different amounts of plant design knowledge; These models were implemented and tested against several real time attack scenarios. Interestingly, DAE-$\boldsymbol{O_{AD}}$ outperforms the other two variants, since each AE model captures the sensor dependencies within the particular stage more effectively and also the computational complexity is minimized due to their distributed nature. Similar observations are reported in [16] when using LSTM based autoencoders.

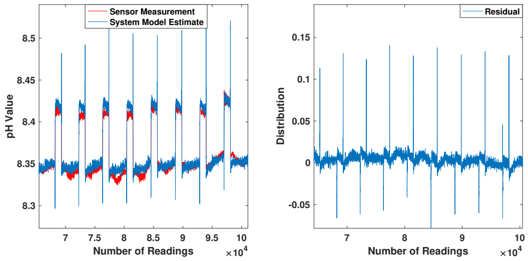

Fig. 2. Training Phase: data used for creating a system model for a pH sensor.

Challenge 3: Scalability: The reference system for this article is the SWaT testbed [19]. There is a multitude of sensors including level, flow, pressure, and chemical sensors for measuring the water quality and quantity. Studies have

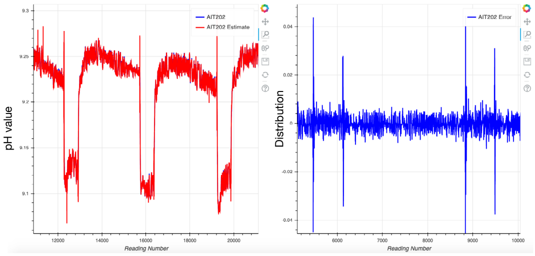

Fig. 3. Testing Phase: validating the model in the process plant.

reported results from using models derived using supervised and unsupervised learning [21] on the SWaT testbed. It has been observed that supervised learning lacks scalability due to the lack of labeled data. On the other hand, unsupervised algorithms can be trained for a large process plant without the need of having a labeled dataset. An interesting example of the scalability of one class classifiers is found in [21] for the case of sensor fingerprinting. The idea is that by using a one-class classifier for each sensor, a unique fingerprint is created to detect intrusion without the need to train the classifier based on the labeled data from all the sensors. The limitation for supervised learning, in that case, is in the event of an increase or decrease in the number of sensors; the models would need to be retrained but using one-class classifier the models would be retrained only for the affected sensors.

Position 4: Unsupervised learning has been shown to better scale in real world ICS.

Challenge 4: Data availability and reliability: Data availability plays a vital role in the design and performance of any anomaly detector. Prior to model creation, one ought to ensure that there exists sufficient amounts of data that represents the components' entire performance cycle and covers all possible modes of ICS operation in the absence of temporal glitches and outliers. The dataset available in [22], represents only one of the SWaT's operational modes. For example, it does not include the behaviour of few backup actuators such as P102, P302, and P402. Though these components are not operational frequently, it is necessary to ensure the presence of their behaviour in the training dataset to guarantee minimal false alarms. Recently researchers [23] conducted a statistical analysis using Kolmogorov-Smirnov test (K-S test) on SWaT, WADI, and the BATADAL datasets to quantify the similarity between the probability distributions of the training and testing data. Outcome of this work has led to the avoidance of several features (ICS components) for model creation since there exists difference between the distribution in training and testing samples. Further, the authors claim that the absence of these features forms an important reason for the reduced false alarm rate of the proposed model.

B. Challenges: Model deployment

Next we point to challenges that arise in the second phase of model development, namely the model deployment phase.

Challenge 5: Data sampling rate : Using a anomaly detectors requires sampling plant data at regular intervals. During deployment of the trained models in the SWaT testbed, the sampling rate was not constant due to the load on the server used for collecting and disseminating plant data. Such a variation in the sampling rates, i.e., different sampling rates during training and deployment, causes the detector to fail to perform as expected.

Challenge 6: System Modeling/Exhaustive discovery of state space at the training phase: Modeling a cyber physical process is important for a variety of reasons. Intrusion detectors use such system models to learn the baseline of a system [24]. System models can be learned either using machine learning methods or subspace system identification techniques. In the latter case, an important task is to determine how stable is the resultant model. For creating a system model, the input is the process data in the form of sensor measurements. It is challenging to get a stable model when the physical quantity/input has been modified. For example, a level sensor in a water tank is not affected by the quality of the water. In other words, the water level should stay within the defined control limits regardless of whether it is clean or dirty water. On the other hand, the quality of water affects the sensor measurements. Thus, when there exist impurities, not present during model creation, then such a change in the data input will raise false alarms.

Figure 2 shows the estimates obtained for a pH sensor based on the system model obtained from the data. It can be seen from Figure 2 that the obtained system model can accurately estimate the sensor measurement but when tested in the live plant the distribution of residuals for this model is radically different. Validation results are in Figure 3. Quality of the incoming water is the prime cause for the observed deviation; during the training phase in the lab environment, pH of the incoming water was in a different range as compared to that during the testing phase.

Position 5: Exhaustively discovering the entire physical state space of a process during the training phase is a challenging problem.

C. Challenges: Model retraining and retuning

Once a detector has been built, tested, validated, and has been in operation in a live plant, it may begin to perform differently than how it did during the early phases of operation. This observation leads to several challenges described next.

Challenge 7: Distribution Shift:

There can be instances when the behavior is physically acceptable for the process but the detector raises alarms. This may happen due to component degradation over time. Another reason relates to the plant modes of operation. Thus, a plant may have multiple modes of operation though data was collected for model creation using a subset of such modes. For example, a power grid may operate in a different mode during public vacation periods than the during normal period.

Position 6: It is challenging to figure out the distribution shifts in the data due to the component degradation or process variations at the time of training a model.

Challenge 8: Noisy data: It has been demonstrated that an attacker can "hide" in the noise distribution of the data [5]. In [25] the authors conclude that often machine learning algorithms miss the attacks in the noisy process data. For such a stealthy attacker it is important to consider the process noise distribution to train the detector.

Position 7: Process noise needs to be considered while modeling the physical process.

The challenge arises because the noise is specific to the particular state of the process. For example, a water tank filling process in a tank would exhibit different noise profile as compared to the water tank emptying process.

Challenge 9: Attack Localization: It has been reported that even though detectors using machine learning algorithms can detect anomalies, they fail to provide hint relevant to the location in the plant where the anomaly may have originated. One solution to this problem is to use a model for each sensor. However, doing so may miss anomalies created due to coordinated multi-point attacks.

Position 8: Locating the source of attack when detectors created using machine learning, is a challenging problem.

Challenge 10: Unbalanced Data: The problem of unbalanced data in the use of machine learning for anomaly detection is well understood [26]. Specifically, the problem is that under normal settings, few examples of anomalous data, as compared to the normal data, are available to train the models. This results in a classifier biased towards normal operation; any slight change due to process noise may therefore be flagged as an anomaly. The problem of unbalanced data due to the lack of attack data exists in the ICS domain. Moreover, there are additional reasons for this imbalance, for example, failure of components and the modes of operation.

Position 9: The problem of unbalanced data-set in ICS goes beyond the legacy IT systems.

Challenge 11: Parameter Alteration: Over the lifetime of a plant, the process parameters might be altered. For example, for a water storage tank there are set points such as high (H) and low (L) that represent normal operating levels. Boundary set points can be high-high (HH) and low-low (LL). Due to sudden increase or reduction in demand or for process optimization due to economical constraints, for example, reducing the consumed power, these process parameters can be altered. This renders useless the models trained during the design stage.

Position 10: Change in process parameters render useless models trained during design stage.

Challenge 12: Model Validation: In terms of data driven approaches, model validation is a process of testing the developed model using the historical dataset prior to its deployment in the plant. During this process, since the validation dataset does not consist of anomalies, the performance of the model is evaluated in terms of false positives. For example, in the case of SWaT, current anomaly detectors such as GARX [27] and MLP_CUSUM approaches were validated with several data subsets collected from the SWaT testbed at different time intervals. Since these models were developed using mostly parametric approaches, for fine tuning of the parameters that captures the dynamic nature of ICS, the validation dataset must effectively reflect multiple modes of operation.

Challenge 13: Model complexity: Model complexity plays a vital role in the selection of the type of learning model to design an anomaly detector. Generally, the complexity of the model refers to the number of hyperparameters that need to be fine-tuned during the training process. This varies based on the type of the model under consideration. For example, in case of MLP_CUSUM, due to the non-linearity of the SWaT dataset, several hyperparameters, namely number of hidden layers, number of input and hidden neurons, learning rate, weight decay, momentum, and dropout factor, were fine tuned during the training process to achieve minimal forecast error. Similarly, for one class SVM, authors in [20] have fined tuned the parameters, namely c and γ for better performance on the SWaT dataset. Although there exists several automated approaches, such as gird search, randomized search, and meta heuristic optimization techniques for fine tuning, a significant challenge we face is with overfitting. Generally, the error rate during the validation process should be less for the trained model; higher validation error for the model trained with large volume of data implies that the model is over-fitted.

Challenge 14: Attack Detection Speed: The speed at which a process anomaly is detected is of prime concern due to safety of the plant. The earlier the anomaly is detected, and reported, the sooner appropriate actions to mitigate the imp[act could be undertaken.

Position 11: The speed of anomaly detection is an important parameter to consider while designing machine learning-based intrusion detection for ICS.

Position 12: Machine learning models must be retrained to cater to environmental effects.

V. Future work and Recommendations

Challenges mentioned above lead to new research directions. Following are a few recommendations for future work based on these challenges.

Recommendation 1: Feature Engineering: From the previous studies on the SWaT dataset, it is observed that most of the research is focused on using machine learning on the raw process data. Taking all the process data and using it as input to machine learning algorithms is susceptible to adversarial attacks as demonstrated in [28]. Therefore, it is important to derive features that are specific to the physical process in an ICS. For example, one study on SwaT used the measurement noise generated from the sensors with machine learning algorithms to design an anomaly detector [21]. *It is recommended that feature engineering should be specific to the physical process for it to be relevant to the ICS.*

Recommendation 2: Define the scope of the IDS: Sommer and Paxson [29] recommended to define the scope of the IDS for the legacy IT systems. In the realm of ICS, this recommendation becomes even more relevant as these are complex systems composed of both cyber and physical components. An IDS for ICS should have a clearly defined scope. It would be challenging to come up with an IDS which could detect cyber, i.e., in the ICS communications network, as well as a physical anomalies.

Recommendation 3: Distinguish between fault and attack: Most of the studies reported using the SWaT testbed have used the process data. It is a challenge to determine whether the reported anomaly is due to a fault or an attack. It is recommended to design a detector which could distinguish between an anomaly due to a fault with that due to an attack [30]. For example, if a sensor reports a measurement that is not expected by the machine learning model, can we determine whether this i anomalous measurement is due to a cyber attack or a fault in the sensor?

REFERENCES

[1] S. Weinberger, “Computer security: Is this the start of cyberwarfare?” *Nature*, vol. 174, pp. 142–145, June 2011.

[2] R. Lipovsky, “New wave of cyber attacks against Ukrainian power industry,” January 2016, http://www.welivesecurity.com/2016/01/11.

[3] P. Cobb, “German steel mill meltdown: Rising stakes in the internet of things,” https://securityintelligence.com/german-steel-mill-meltdown-rising-stakes-in-the-internet-of-things/, 2015.

[4] S. Adepu and A. Mathur, “Distributed attack detection in a water treatment plant: Method and case study,” *IEEE Transactions on Dependable and Secure Computing*, pp. 1–8, 2018.

[5] C. M. Ahmed, J. Zhou, and A. P. Mathur, “Noise matters: Using sensor and process noise fingerprint to detect stealthy cyber attacks and authenticate sensors in CPS,” in *Proceedings of the 34th Annual Computer Security Applications Conference, ACSAC 2018, San Juan, PR, USA, December 03-07, 2018*, 2018, pp. 566–581.

[6] P. V. Overschee and B. D. Moor, “Subspace identification for linear systems: theory, implementation, applications,” *Boston: Kluwer Academic Publications*, 1996.

[7] A. P. Mathur and N. O. Tippenhauer, “SWaT: A water treatment testbed for research and training on ICS security,” in *International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater)*. USA: IEEE, April 2016, pp. 31–36.

[8] iTrust, “iTrust Datasets,” https://itrust.sutd.edu.sg/itrust-labs_datasets/.

[9] G. Sugumar and A. Mathur, “Testing the effectiveness of attack detection mechanisms in industrial control systems,” in *2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, July 2017, pp. 138–145.

[10] F. Zhang, H. A. D. E. Kodituwakku, W. Hines, and J. B. Coble, “Multilayer data-driven cyber-attack detection system for industrial control systems based on network, system and process data,” *IEEE Transactions on Industrial Informatics*, 2019.

[11] R. Mitchell and I.-R. Chen, “A survey of intrusion detection techniques for cyber-physical systems,” *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, p. 55, 2014.

[12] D. Shalyga, P. Filonov, and A. Lavrentyev, “Anomaly detection for water treatment system based on neural network with automatic architecture optimization,” in *ICML Workshop for Deep Learning for Safety-Critical in Engineering Systems*, 2018, p. 1?9.

[13] J. M. Beaver, R. C. Borges-Hink, and M. A. Buckner, “An evaluation of machine learning methods to detect malicious scada communications,” in *2013 12th International Conference on Machine Learning and Applications*, vol. 2. IEEE, 2013, pp. 54–59.

[14] R. C. B. Hink, J. M. Beaver, M. A. Buckner, T. Morris, U. Adhikari, and S. Pan, “Machine learning for power system disturbance and cyber-attack discrimination,” in *2014 7th international symposium on resilient control systems (ISRCS)*. IEEE, 2014, pp. 1–8.

[15] S. Priyanga, M. Gauthama Raman, S. S. Jagtap, N. Aswin, K. Kirthivasan, and V. Shankar Sriram, “An improved rough set theory based feature selection approach for intrusion detection in scada systems,” *Journal of Intelligent & Fuzzy Systems*, vol. 36, no. Preprint, pp. 1–11, 2019.

[16] M. Kravchik and A. Shabtai, “Detecting cyber attacks in industrial control systems using convolutional neural networks,” in *Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy*. ACM, 2018, pp. 72–83.

[17] J. Goh, S. Adepu, M. Tan, and Z. S. Lee, “Anomaly detection in cyber physical systems using recurrent neural networks,” in *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*. IEEE, 2017, pp. 140–145.

[18] J. Inoue, Y. Yamagata, Y. Chen, C. M. Poskitt, and J. Sun, “Anomaly detection for a water treatment system using unsupervised machine learning,” in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2017, pp. 1058–1065.

[19] A. P. Mathur and N. O. Tippenhauer, “Swat: a water treatment testbed for research and training on ics security,” in *2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater)*, 4 2016, pp. 31–36.

[20] J. Inoue, Y. Yamagata, Y. Chen, C. Poskitt, and S. Jun, “Anomaly detection for a water treatment system using unsupervised machine learning,” in *IEEE International Conference on Data Mining Workshops*. IEEE, 2017, pp. 1658–1065.

[21] C. M. Ahmed, M. Ochoa, J. Zhou, A. Mathur, R. Qadeer, C. Murguia, and J. Ruths, “Noiseprint: Attack detection using sensor and process noise fingerprint in cyber physical systems.” in *Proceedings of the 2018 ACM on Asia Conference on Computer and Communications Security*, ser. ASIA CCS ’18. ACM, 2018.

[22] J. Goh, S. Adepu, K. Junejo, and A. Mathur, “A dataset to support research in the design of secure water treatment systems,” in *International Conference on Critical Information Infrastructures Security*. IEEE, 2016, p. 88?99.

[23] M. Kravchik and A. Shabtai, “Efficient cyber attacks detection in industrial control systems using lightweight neural networks,” *arXiv preprint arXiv:1907.01216*, 2019.

[24] C. M. Ahmed, C. Murguia, and J. Ruths, “Model-based attack detection scheme for smart water distribution networks,” in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ser. ASIA CCS ’17. New York, NY, USA: ACM, 2017, pp. 101–113. [Online]. Available: http://doi.acm.org/10.1145/3052973.3053011

[25] 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.

[26] G. Ho, A. Sharma, M. Javed, V. Paxson, and D. Wagner, “Detecting credential spearphishing in enterprise settings,” in *26th {USENIX} Security Symposium ({USENIX} Security 17)*, 2017, pp. 469–485.

[27] J. Heng and Y. C. Huei, “Machine learning invariants to detect anomalies in secure water treatment (swat),” in *International Conference on Machine Learning and Communications Systems(ICMLCS 2019)*, January 2019, pp. 129–136.

[28] G. Zizzo, C. Hankin, S. Maffeis, and K. Jones, “Invited: Adversarial machine learning beyond the image domain,” in *2019 56th ACM/IEEE Design Automation Conference (DAC)*, June 2019, pp. 1–4.

[29] R. Sommer and V. Paxson, “Outside the closed world: On using machine learning for network intrusion detection,” in *2010 IEEE symposium on security and privacy*. IEEE, 2010, pp. 305–316.

[30] D. E. Denning, “An intrusion-detection model,” *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 222–232, Feb 1987.