Machine Learning for Intrusion Detection in Industrial Control Systems: Applications, Challenges, and Recommendations

Muhammad Azmi Umer
DHA Suffa University
Karachi Institute of Economics and Technology
muhammadazmiumer@yahoo.com

Khurum Nazir Junejo
DNNae Inc.
junejo@gmail.com

Muhammad Taha Jilani
Karachi Institute of Economics and Technology
m.taha@kiet.edu.pk

Aditya P. Mathur
Singapore University of Technology and Design
aditya_mathur@sutd.edu.sg

ABSTRACT

Methods from machine learning are being applied to design Industrial Control Systems resilient to cyber-attacks. Such methods focus on two major areas: the detection of intrusions at the network-level using the information acquired through network packets, and detection of anomalies at the physical process level using data that represents the physical behavior of the system. This survey focuses on four types of methods from machine learning in use for intrusion and anomaly detection, namely, supervised, semi-supervised, unsupervised, and reinforcement learning. Literature available in the public domain was carefully selected, analyzed, and placed in a 7-dimensional space for ease of comparison. The survey is targeted at researchers, students, and practitioners. Challenges associated in using the methods and research gaps are identified and recommendations are made to fill the gaps.

KEYWORDS

Machine Learning, Deep Learning, Intrusion Detection, Anomaly Detection, cyber-attacks, Cyber Physical Systems, Critical Infrastructures, IoT, Industrial Control Systems

1 INTRODUCTION

This article is a survey of methods from machine learning (ML) that are being applied to detect intrusions, or anomalies, in systems. The systems of interest in this survey are primarily those where an Industrial Control System (ICS) is used to control a physical process. Such systems are constituents of critical infrastructure in a city and country, and include the electric power grid, water treatment and distributions systems, and oil refineries. Such systems are a subset of a broader class of systems known as Cyber-Physical Systems (CPS) that consist of cyber and physical subsystems. These subsystems are integrated via sensors, actuators, and communications links to enable the control of the underlying physical process [20, 21, 158]. While ICS remain the focus of this survey, we have not avoided references to systems that do not use ICS, but fall in the CPS category.

Industrial Control Systems: ICS include a Supervisory Control and Data Acquisition (SCADA) system, Programmable Logic Controllers (PLCs), Remote I/O (RIO) units, sensors, and actuators. While the specific brand and types of such subsystems may differ, their overall function is to effectively control the underlying physical process. Successful and unsuccessful attempts to affect the behavior of ICS has led to an increase in research aimed at developing methods

and tools to protect plants from malicious actors [140, 166]. Such attempts by malicious actors are made possible, and are sometimes successful, due to a variety of reasons including inadequate physical and or cyber protective measures and network connectivity.

Attacks on ICS: Data in Table 1 is indicative of the rise in successful cyber-attacks on ICS. A uranium enrichment plant in Iran was attacked [44] resulting in an increase in the failure of centrifuges. The Maroochy water services were attacked by an ex-employee and a large quantity of sewage spilled into a local park [174]. A water treatment plant in the U.S. was attacked in 2006 [31]. Such attacks, and their impact, has led to a realization that new methods and tools, beyond the traditional mechanism, e.g., firewalls that protect communication networks, are needed to protect ICS.

Target audience: Given an increasing body of literature focusing on using ML for defending ICS against cyber-attacks, it is important to subject this body of work from a critical perspective for the benefit of researchers, students and practitioners. Researchers and students aiming to explore the use of ML in defending ICS against cyber-attacks stand to benefit from this survey as it would allow them to identify gaps in the literature and weaknesses of existing methods. Practitioners, aiming to develop commercial tools for use in operational plants, stand to benefit from this survey as it would help them identify the most promising methods on which to base their tools.

Keeping the survey live: Given the rate at which research is progressing in the application of machine learning to detect cyber intrusions, it is likely that this survey will rapidly be rendered incomplete, or even outdated, soon after its publication. To ensure that the survey remains up-to-date, we have created a web site[1] where we will add new literature in this area with suitable comments. Tables in this article that place each research publication in a 7-dimensional space will be kept at this site and updated regularly.

Abbreviations and nomenclature: Given the focus of this survey, the terms "plant," "system," and ICS are used synonymously. Such usage is justifiable as an ICS is a subsystem in a physical system and, when attacked, it impacts the underlying process, e.g., water filtration or uranium enrichment. We note that ICS enabled systems are Cyber-Physical Systems. However, as much as possible, we have avoided the use of the term CPS due its breadth and the fact that

---

[1]https://sites.google.com/view/crcsweb/survey-paper

literature surveyed here focuses mostly on plants controlled by an ICS. Literature related to detection of anomalies in network traffic is generally classified under "Intrusion detection" category. However, literature in the ICS domain that focuses on physical processes in a plant, is classified under "anomaly detection." In this survey we use the "intrusion detection" to refer to anomaly detection in physical plants as well as the detection of network intrusions. Techniques from machine learning are often referred to by their abbreviations, e.g., RNN for Recurrent Neural Networks. This survey uses a large number of such abbreviations. To make it easy for a reader new to machine learning each abbreviation used in this article, and its expansion, is listed in alphabetic order in Table 8 placed at the end of this survey.

Organization: The remainder of this survey article is organized as follows. In Section 2 we introduce Intrusion Detection Systems (IDS) and categorize them broadly. A large number of articles had to be collected for this survey to be possible. The collection process is summarized in Section 3. There are other surveys reported that also focus on ML techniques as applied to ICS. Such surveys are cited with differences from our survey identified in Section 4. The literature surveyed and evaluated is placed in a 7-dimensional space described in Section 5. Various methods from machine learning used for intrusion detection are categorized and explained in Section 6. This is followed by Sections 7, 8, 9, and 10 where we examine, respectively, the literature that focuses on the use of supervised, unsupervised, semi-supervised, and reinforcement learning for intrusion detection. Major challenges and recommendations related to IDS in ICS are discussed in Section 11. Section 12 has summarized the overall work and discussed the conclusion.

2 INTRUSION DETECTION SYSTEMS

Before diving into a detailed survey, we summarize below the various types of intrusion detection systems (IDS). Such systems aim at detecting intrusions and anomalies during plant operation. The detected intrusions and anomalies are reported to plant engineers who are then expected to take appropriate actions to prevent undesirable consequences such as service disruption and component damage. Three types of IDS are considered in the following, namely, signature-based, specification-based, and behavior based.

2.1 Signature-based IDS

This type of IDS requires a predefined dictionary of attack patterns. It detects an intrusion if any pattern detected during plant operation matches one or more of the predefined attack patterns [53]. Though this approach maintains a low rate of false positives, it fails to detect zero-day attacks. Further, it is often difficult to produce an exhaustive dictionary of attack signature in complex physical processes. There are numerous ways to automate the generation of malware signatures. For example, in the study reported in [137] malware signatures were generated in private cloud using deep feature transfer learning. Volatile memory dumps were extracted during the malware activity by querying the hypervisor of the virtual machine. Malicious processes were extracted from the memory dumps and converted to images. Later, these images served as input to a pre-trained deep neural network model, namely, VGG19. The proposed model is robust and fast as it does not require training on new input data. However, as it generates signatures using only the available malware processes, it could be prone to zero-day attacks.

2.2 Specification-based IDS

This approach develops a mathematical model to define the normal operation of the physical process under consideration. An anomaly is said to exist whenever the process deviates from the prediction by the predefined model [130]. Such models are developed with the help of experts and plant design. While the experts may have knowledge of physical processes, there are issues related to the aging of the physical system, inaccuracies that may exist in operational manuals, and interpretation of the process behavior. Secondly, it is difficult to develop accurate mathematical models for complex distributed physical systems. The study reported in [2] derived the invariants (specifications) from the design of a water treatment plant. They used it to detect cyber-attacks on the plant. However, unless automated, the proposed approach is unable to derive the specifications of complex physical processes that are not reflected in the design document. A study reported in [24] also used a specification-based approach for intrusion detection in Advanced Metering Infrastructures (AMI). They used sensors to monitor the traffic at meters and access points at the network, transport, and application layer. They made a set of specifications and policies to ensure the safety of meters and AMI, respectively.
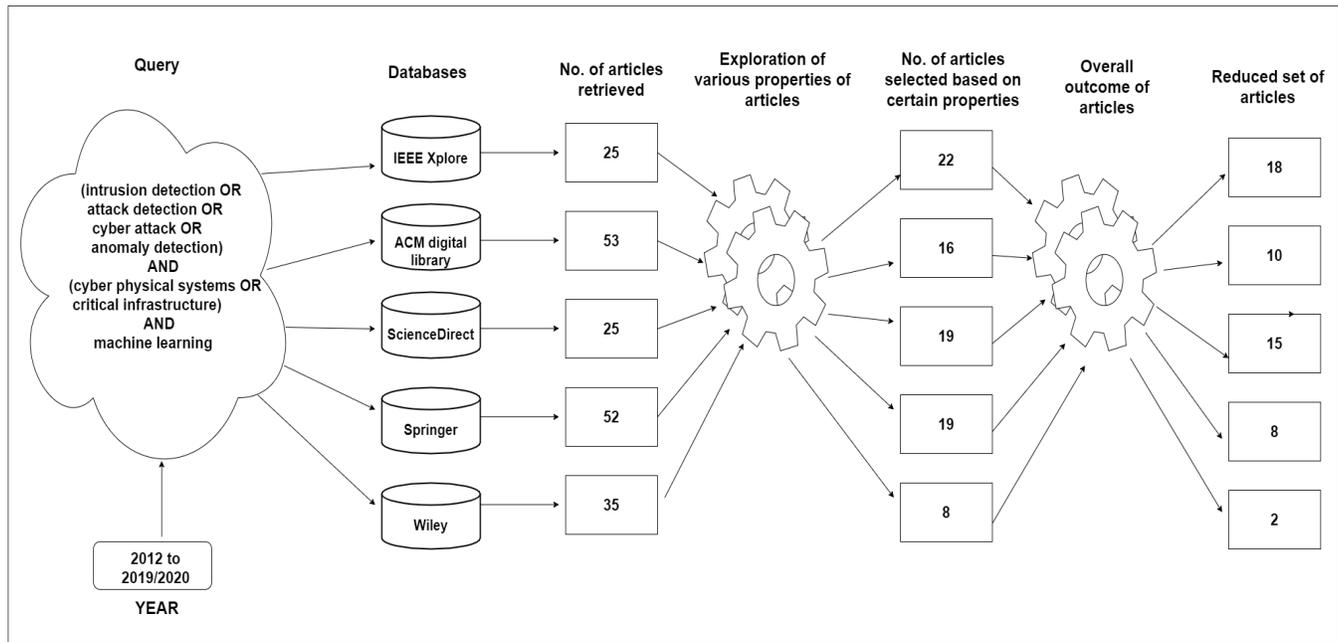
2.3 Behavior-based IDS

This approach is based on the operational data from the physical system. Based on data collected, a model is trained on the normal and abnormal behavior of the process and used to detect intrusions. This approach is favored against incorrect vendor specifications as it trains the model on empirical data [79] and thus helps in identifying incorrect vendor specifications. For instance, a study reported in [82] noticed different levels of a water tank in a water treatment plant. According to the vendor specifications, the upper bound of on the volume of water in the tank was 1100 liters; this value was also encoded in the control logic of the PLCs in the ICS. Analysis of data obtained through level sensors associated with the tank revealed that the upper bound in practice was 900 liters.

Traditional behavior-based approaches relied on statistical techniques [206] such as the mean and standard deviation of sensor readings. Lately, machine learning (ML) techniques are being used extensively as behavior-based approaches to secure ICS. State of the art techniques using this approach have been reported in the literature. Such techniques are gaining popularity among researchers and commercial vendors mainly due to the availability of high computing power and tools to detect the non-linear relations and unobserved regularities in the massive volumes of data. Nevertheless, there remain serious problems associated with these techniques including the detection of zero-day attacks, ensuring an acceptable rate of false alarms, and managing computational complexity. These problems are creating a bottleneck for the deployment of IDS based on these techniques, in particular in complex Industrial Control Systems (ICS). This article discusses these techniques in detail within the paradigm of intrusion detection in ICS. It also discusses the associated problems and offers recommendations.

Table 1: Incidents on Industrial Control Systems

Year	Incident	Year	Incident
2019	LockerGoga Ransomware [124]	2014	Port Hudson Paper Mill Insider Threat [187]
2018	Olympic Destroyer [63]	2013	Havex [155]
2018	TRITON Triconex SIS Malfunction [161]	2012	Shamoon [154]
2017	TEMP.isotope Campaign [139]	2011	Duqu [62]
2017	BadRabbit Ransomware [142]	2010	Stuxnet [44]
2017	EternalPetya Ransomware [123]	2008	CIA Reports Foreign Utilities Hacked [122]
2017	WannaCry Ransomware [50]	2007	Aurora Generator Test [75]
2016	Industroyer Ukraine Blackout [143]	2003	Northeast Blackout [127]
2015	BlackEnergy 3 Ukraine Blackout [110]	2001	Maroochy Sewage Spill [174]

Figure 1: Retrieval and Selection of Articles

3 COLLECTION OF ARTICLES

Apart from other relevant articles, a major set of articles reported in this study were collected using a systematic approach. Due to the inaccessibility of Web of Science and Scopus, five major databases including IEEE Xplore, ACM digital library, ScienceDirect, Springer, and Wiley were explored in-depth. Several queries were used to retrieve the relevant articles. These queries can be combined to form a single query using logical connectives, as for example *(INTRUSION DETECTION OR ATTACK DETECTION OR CYBER ATTACK OR ANOMALY DETECTION) AND (CYBER PHYSICAL SYSTEMS OR CRITICAL INFRASTRUCTURE) AND MACHINE LEARNING.*

All articles from 2012 to 2020 were retrieved in multiple iterations as described in Figure 1. In the first iteration, a breadth-first study of each article was performed to extract various properties including the approach, limitations, strengths, etc. In the second iteration, articles were selected based on the relevance of the proposed approach to ICS. For example, some articles were related to IDS but did not emphasize ICS or CPS, and hence were not selected for further analysis. Twenty-five articles were retrieved from IEEE Xplore. Here, the focus was only on journals and magazine articles. Fifty-three articles were retrieved from the ACM Digital Library. From this library, only the articles from journals and conferences of core rank A and B were selected. Twenty-five articles were retrieved from ScienceDirect. Nineteen articles were selected in the first iteration and fifteen in the second. Fifty-two articles were retrieved from Springer of which nineteen articles were selected in the first iteration and eight in the second. Thirty-five articles were retrieved from Wiley of which eight were selected in the first iteration and only two were selected in the second.

Table 2: Comparison with Past Surveys

Past Surveys	Difference
[26]	Main theme is to shift current ICS to cloud based infrastructure.
[12]	Focus on IDS in general terms; not specifically on ICS.
[115]	Focus on Deep Learning (DL) techniques with types of anomalies, evaluation metrics, strategies, and implementation details; different taxonomy
[60]	A general survey of physics-based attack detection in CPS; not focused on ML.
[129]	A survey of IDS in CPS focusing only on detection technique and audit material.
[65, 172]	A survey of CPS discussing challenges and future trends; does not focus on IDS approaches for CPS.
[15, 19, 35, 54, 178]	Focus on Network-based IDS.
[11]	Focus on Reinforcement Learning (RL) based Q-learning methods for securing CPS.
[216]	Focus on SCADA specific intrusion detection and prevention.

4 RELATED SURVEYS

A comparison of related surveys is presented in Table 2. A survey of ICS security focusing mainly on ML is reported in [26]. The article has discussed the benefits and shortcomings of using ML techniques for detecting anomalies in ICS. The need for shifting current ICS to cloud-based infrastructure was the main theme of this research. This survey has discussed minimal work on machine learning-based IDS. Also, only an overview of machine learning approaches was emphasized.

Deep Learning-based intrusion detection systems are discussed in [12]. This work focuses on intrusion detection in its general terms, not focusing on ICS. The work is divided into the frameworks, developed IDS, datasets, and testbeds. A survey of deep learning techniques for anomaly detection is reported in [115]. A taxonomy was developed for the survey which includes type of anomalies, evaluation metrics, strategies, and implementation details.

A survey of physics-based anomaly detection is reported in [60]. The authors developed a taxonomy to identify the key characteristics of their survey. This taxonomy consists of attack detection, attack location, and validation. Attack detection is divided into prediction and detection statistics. Metrics and the implementation to verify and validate the performance of attack detection algorithms, are discussed. A survey of intrusion detection techniques is reported in [129]. This survey focuses on two dimensions, i.e., the audit material and detection techniques. Apart from these two dimensions, the survey reported in the article here focuses on several other dimensions as well as discussed in section 5.

A survey of CPS is reported in [65, 172]. This survey discusses the challenges and future research trends but did not focus on IDS approaches for CPS. The network-based IDS was surveyed in [15, 19, 35, 54, 178], but the authors do not address the scenario which differs from conventional networks. A survey reported in [11] focuses on reinforcement learning based Q-Learning method for securing a CPS. The survey focused on CPS in terms of supported techniques, domains, and attacks. The study reported in [216] focused on SCADA specific intrusion detection and prevention. The survey presented in this article focuses on behavior-based approaches for intrusion detection in CPS focusing on ML and DL techniques. Recently these approaches have gained more attention as they are relatively easier to automate than others, and are scalable and generalizable for new ICS.

5 DIMENSIONS FOR CLASSIFYING INTRUSION DETECTION SYSTEMS

Recent progress in ML coupled with attempted and successful cyber-attacks on critical infrastructure, has sparked a wave of interest in behavior-based IDS for ICS. It is important for researchers and practitioners to understand how the proposed approaches compare with each other and their usability in operational environments. With this as our goal, it was decided to adopt a multi-dimensional approach to categorize the literature most of which focuses on IDS for ICS while some on a broader class of CPS. Specifically, works surveyed in this article are placed in a 7-dimensional space where the dimensions are domain, audit material, complexity, feature selection, time series, dataset, and metrics. The use of this multi-dimensional space adds formalism to the comparison of different works and enables a scientific discussion on their utility or non-utility in specific environments. We mote that the adoption of a multi-dimensional approach for categorization of research has also been adopted by other researchers [129]. However, the multi-dimensional space adopted by us is richer in terms of the dimensions selected and their number. The dimensions used in the work are enumerated in Table 3 and described in the following subsections.

5.1 Domain

Intrusion detection for ICS has been applied in a variety of domains, including smart utilities. Not surprisingly, most applications are in the area of energy, water and gas primarily because of the critical nature of these systems. A power grid compromised for a few seconds can trip a generator. This transfer may result in the affected load transferred to other generators and possibly initiate a cascade of generators tripping one after the other leading to a major blackout. The works labelled as Annon in Table 4, 5, 6, and 7 do not specify the domain on which the proposed approach is applied, instead they mention it as "some CPS/ICS".

In our survey we found that the least explored ICS in smart utility is gas. Even though a few such ICS are listed in Table 7, they rely

Table 3: Dimensions used for categorizing literature in this survey.

Dimension	Description
Domain	Application domain such as electric power grid and water treatment plant.
Audit material	Data used in model creation
Complexity	Computing power needed; scalability
Feature selection	Selection of features to reduce overfitting
Time series	Modeling processes as a time series
Dataset	Data used; pre-collected or live; from simulation or live plant
Metrics	Metrics used for evaluating the effectiveness of the ML techniques used

on a relatively simple gas ICS testbed at Mississippi State University (MSU) [131], which consists of a a minimal set of components including pressure sensor, a pump, and a solenoid valve.

5.2 Audit Material

Typically the data analyzed by an IDS includes network traffic and sensor measurements with few IDS considering both. Since IDS were first developed for the internet and LAN networks, most of the IDS developed for ICS also attempt to detect intrusions in the network layer using similar approaches. Typically, ICS use industrial control protocols such as Modbus [37], BACnet [165], and DNP3 [25]. Hence, it is commercially viable to develop IDS for such protocols. A study reported in [97] used bits per packet, connections per second, and recent/mean interval time and count of Goose messages for this purpose. Another study reported in [52] used several responses against a command to detect attacks. The study in [40, 66] did deep packet inspection to calculate the n-gram features from the payload of the packet. The method used in this work is to constructs a feature vector that contains the count, frequency, and binary occurrence of these n-grams. The authors also argue that n-grams are successful in detecting attacks. However, the approach proposed in [165] uses Ethernet, IP, UDP, and BACnet packet header attributes to train its IDS. The study reported in [120] suggests detecting attacks by using the number of live TCP, UDP & ICMP connections, duration of terminated connections, overall network fragments pending reassembly by Bro, amount of data sent by connection responder/originator, and the number of packets sent by connection responder/originator features.

Detecting attacks in the physical process controlled by an ICS is challenging as components, size, and functionality of each process is different from others. Such IDS have received relatively little attention and though at the time of writing this survey there seems to be a growing trend to detect intrusion at the physical process level. IDS that model the physical process of the energy systems have used the following features to train their model: voltage Phase angle, voltage magnitude, current phase angle, current phase magnitude, zero voltage phase angle magnitude, current phase angle magnitude, the frequency of relays, frequency delta for relays, apparent impedance seen by relays, angle seen by relays, status flags for relays, snort alert status for each relay, control panel remote trip status, and their correlations [28, 148]. A study reported in [82] used the status of the pumps and valves, rate of inflow, level of the tank, and rate of change of water level for water ICS. For gas ICS, [134] uses pressure in the pipeline, pump, and solenoid status as features.

There have been few attempts in developing a hybrid approach by using both the network traffic and physical process features. A study reported in [23, 52] used a couple of physical process features along with a few dozen network traffic features to detect attacks in gas ICS. Also, a study reported in [205] used CPU and OS usage parameters in addition to features of network traffic to detect attacks in a simulated CPS made up of different SUN Microsystem servers and workstations. A study reported in [92] used Wireshark to capture network logs and physical stream data such as temperature and airflow. This data was then used to learn an IDS for Heating, ventilation, and air conditioning (HVACs). The above-mentioned hybrid approaches have used a single algorithm to model both the network traffic and physical processes.

5.3 Complexity

Based on complexity, we refer to some approaches as simple when they follow the traditional ML life cycle, i.e., derive some features, followed by some feature selection, and training a classifier. Hybrid approaches follow a more complex life cycle by either a) transforming the input features to a transformed features space where a classifier is trained to give better performance [165], or b) multiple classifiers are trained separately but cooperate to arrive at a decision [98, 165, 199]. A study reported in [147] first used the K-means to cluster the data followed by self-organizing maps (SOM) to do the final classification. Another study reported in [98] learns five different SVM's and uses an ensemble of them to detect attacks. Likewise, a three-tier system for state monitoring of a CPS was proposed in [199]. The first tier consists of a threshold-based alarm. The minimum and maximum bound of each sensor are defined here. Anything above or below this bound triggers an alert. The second tier uses a self-organizing fuzzy logic system. The purpose of this layer is to detect anomalies. This tier learns the rules of the CPS itself. The third tier uses an artificial neural network (ANN) to forecast the value of each sensor based on the historical data. Finally, fuzzy logic is used to raise an alarm based on outputs of tier 2 and tier 3. A study reported an anomaly-based IDS for SCADA [147]. It extracts the time correlation between different packets using histograms, followed by Bayesian inferencing, to identify attacks. An alarm is raised if the probability of belonging to anyone of the seen categories is below a speffic threshold.

5.4 Feature Selection

Feature selection techniques are used to increase the accuracy and to reduce the overfitting and training time of the model; the selection could be manual or automatic. Feature selection techniques include

Univariate Selection, Feature Importance, and Correlation Matrix with Heatmap [168]. Deep Learning techniques do not require explicit feature selection because they have an inherent capability to select the best features for the model. A study reported in [138] proposed a feature selection method based on Tabu Search and Random Forest. They used Tabu Search for searching and Random Forest as a learning algorithm for intrusion detection.

5.5 Time Series

Time series data contains a well-defined time pattern consisting of a specific sequence of measurements. This property is quite useful as it helps determine which particular algorithm, such as time series analysis or any other, would be better to apply in the ML or DL model. A study reported in [83] used fuzzy logic to classify the time series data of sensors in CPS. It represented the time series using the distribution of its data samples. This was done using its proposed Intervals Numbers technique. Moreover, the effectiveness of the proposed approach was tested using a benchmark classification problem.

5.6 Dataset

A major bottleneck in the use of supervised ML and DL techniques is the lack of attack data. The attacks on real-world systems are rare and sparse. Therefore, studies that have used actual data from CPS have resorted to simulated attacks to train and evaluate its classifiers [98, 109], thus making the realism and fidelity questionable. Other works resort to validate their model on completely simulated data [28, 148, 173, 205]. Some studies have even used the NSL-KDD99 dataset [22] to validate their IDS whereas, this data is a collection of simulated raw TCP dump data over nine weeks on a military local area network. It is a benchmark dataset for IDS in normal LAN traffic but not for CPS network traffic.

A publicly available dataset is provided by a Critical Infrastructure Protection Center at Mississippi State University (MSU) [1]. Their power system dataset is a simulated smart grid data consisting of data under normal behavior, attacks, and faults. This dataset was used by [28, 148] for intrusion detection in ICS using ML approaches. Their water storage tank and gas pipeline dataset were developed using small scale laboratory testbed and were used by [23, 134, 135] to detect intrusion in CPS using ML approaches. Their water testbed consists of a water tank having a storage capacity of 2 liters, a pump, and a level sensor. It consists of a physical process attribute for the level of the tank and the status of the pump. Apart from that, they have seventeen different network traffic and PLC status attributes. The gas pipeline dataset consists of twenty-three network attributes and PLC status attributes, and only three physical process attributes, namely pressure in the gas pipeline, solenoid, and pump status. Both of these datasets are flawed for ML research as acknowledged by the authors themselves. SWaT dataset [61] is another publicly available dataset of a water treatment testbed. This testbed is an industrial scaled-down replica of a water treatment plant. It has six stages and can produce five gallons per minute of filtered water. Data collection was done by running the plant non-stop for eleven consecutive days. For the first seven days the plant was run in a normal state while during the last four days specifically

crafted attacks were launched on the plant. Therefore, this dataset contains both the normal and attack data of a real testbed. Both network and physical process data were collected for this purpose. Following the publication of the dataset in [61], iTrust has made public several other datasets collected from the SWaT testbed [77]. The SWaT datasets have been used in a large number of research projects including, though not limited to, [76, 79, 80, 190–192].

5.7 Metrics

Intrusion detection is a skewed class problem, also known as class imbalance. This refers to a setting where most of the data belongs only to a single class, e.g., instances of normal behavior in an IDS dataset constitute more than 90% of the dataset. Hence any naive classifier that labels each instance as normal will get an accuracy higher than 90%. Therefore, accuracy is not enough to assess the performance of IDS, and yet some studies only report accuracy (or error graphs). Similarly, some studies report only the detection rate (DR), which is the same as recall. The recall alone is not enough to assess the performance of IDS, as there is a trade-off between precision and recall. A 100% recall can always be achieved by compromising the precision of the system.

For proper evaluation of the effectiveness of an IDS, more than one of the following metrics should be reported: accuracy, precision, recall, F-measure, receiver operating characteristic (ROC), and area under the ROC curve (AUC). Precision measures the correctness of the classifier based on the detection of an attack. A high value of precision leads to a lesser number of false positives (FP). Whereas, recall is the number of attacks detected by the classifier. A high value of recall leads to a lesser number of false negatives (FN). An ideal classifier should have high precision and recall. F-Measure helps us to combine both into a single metric, which is the harmonic mean of precision and recall. It is a more conservative measure than the arithmetic mean of the two. These measures are defined as follows.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$
$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$
$$F - Measure = \frac{2 * Precision * Recall}{Precision + Recall}$$

where TP is the number of attacks correctly classified by the classifier, and TN the number of normal instances classified as normal.

ROC curve is a true positive rate (TPR) plotted against false positive rate (FPR) thresholded at various settings, whereas AUC is the area under this ROC. These measures are considered to be more robust for highly skewed problems [156]. The reason being that by increasing and decreasing the sensitivity of a sensor, the output of the classifier can often be tweaked to make it more (or less) conservative thus achieving a trade-off between FP and FN. The AUC measure allows selecting possibly optimal models by evaluating the performance of the classifier by varying the threshold that decides whether the instance is an attack or not. Unfortunately, few researchers [82, 87, 141, 152, 198] have used this measure for evaluating their respective classifiers.
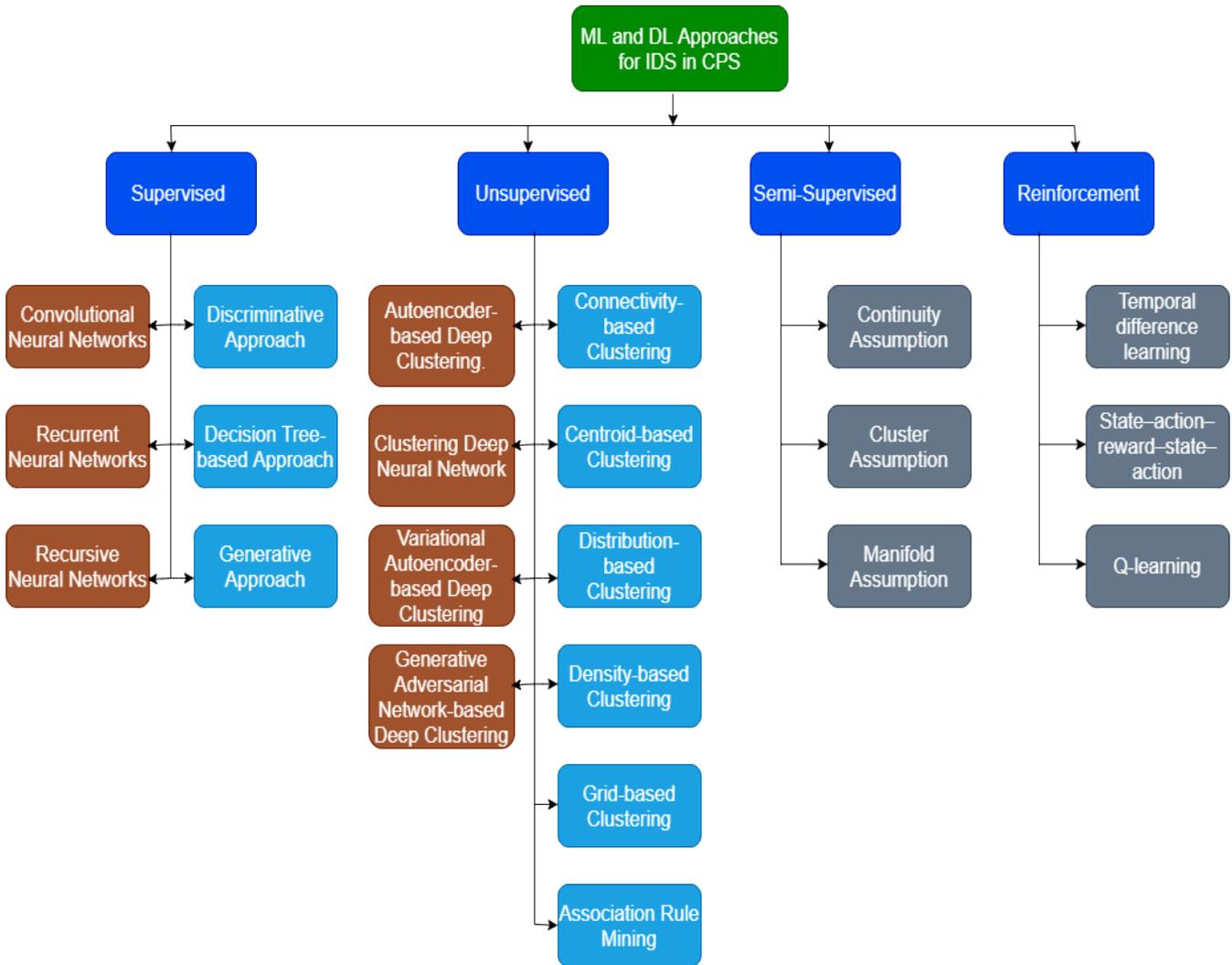
---

[1]https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets

Figure 2: Categorization of machine learning approaches for detecting intrusions in Industrial Control Systems.

The time to detect an attack and the percentage of the time the attack remains detected should also be used as an evaluation metric. It is likely not of any value when an attack is detected once it has already damaged a physical component or the attack is detected intermittently by turning on and off the alarm after every few seconds leading to confusion. Few studies report these measures [82, 98, 191]. Among these, [98] reported the latency. They define latency as the number of cycles after the spoof begins but before the classifier correctly identifies a string of 30 consecutive cycles as spoofed. Whereas [82, 191] reported the time to detect an attack. While [82] also reported the percentage of time the attack was detected during its course.

6 MACHINE LEARNING APPROACHES FOR INTRUSION DETECTION

As shown in Figure 2, ML and DL techniques can be classified into four major categories, i.e. Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, and Reinforcement Learning. Most of the intrusion detection work available in the literature is related to the first two areas while limited work is available in the last two areas. The difference between the first three approaches lies in whether or not the training data used is labeled. An unsupervised approach does not require labeled data, relying solely on the normal behavior of the ICS. A supervised approach requires training data under both normal and abnormal (attack) behavior. The semi-supervised approach makes use of both, relying on the assumption that labeled training data is scarce and rare whereas unlabeled training data is plenty and easily available. All areas mentioned in Figure 2 are discussed in subsequent sections.
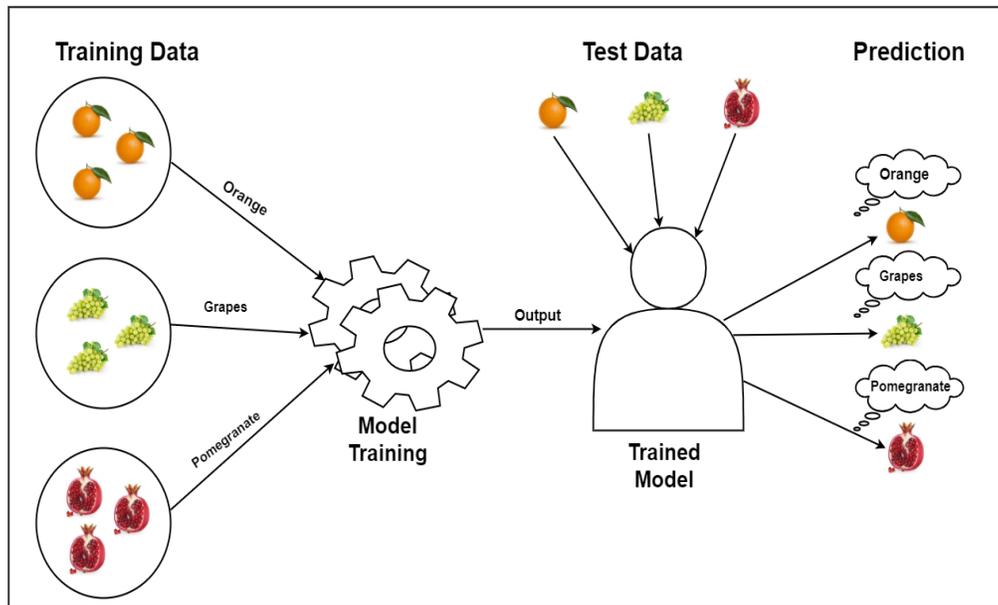
Figure 3: Supervised Learning

Each approach mentioned above has its pros and cons. Unsupervised learning does not require labeled training data, therefore, the dependency on attack data gets eliminated making it capable of detecting zero-day attacks. However, it usually produces high false alarms [135, 136]. While the supervised learning algorithms are more robust in terms of attack detection, they require labeled data, i.e., both normal and attack data. Given only a few instances of attacks, the supervised approach is capable of detecting other instances of attacks as well. The study reported in [82] showed that their best classifiers produce almost no FPs and achieved high precision and recall. These approaches do not have any guarantee in detecting the zero-day attacks.

Another set of promising approaches that have not been explored for IDS in ICS are one-shot learning [93, 200] and zero-shot learning [163, 175]. One-shot learning refers to a scenario where only one instance of each attack type is available in the labeled training data. Whereas zero-shot is a more challenging approach in which few instances of some attacks are available in the labeled training data. The attack type that does not have any instance in the training data represents zero-day attacks. Thus, the performance of this type of learning is based on detecting the zero-day attacks while leveraging the information provided by the known attacks. This represents a more practical approach for an ICS as it would generate fewer FPs than unsupervised approaches and at the same time detect zero-day attacks while leveraging on some known attacks that can be safely carried out on the ICS in a controlled environment. We believe that zero-shot learning is a promising approach for IDS in ICS because it achieves a good compromise between the supervised and unsupervised approaches.

7 SUPERVISED LEARNING

Supervised Learning (SL) requires labeled training data as described in Figure 3. For each instance of the training dataset, SL uses 'n' features from feature vector 'X', i.e., $[x_1, x_2 ....,x_n]$ to learn the class variable 'Y' against each instance of the dataset. The relationship between 'X' and 'Y' is captured in the equation $Y = f(X)$ where f is learned from data.

There are mainly two types of SL techniques: classification and regression. In classification, the class variable is discrete while for regression problems it is continuous. IDS are typically modeled as classification problems where the class variable can contain both single and multiple classes. If the class variable contains only a single class then it is referred to as a One-Class Classification (OCC) problem. OCC-based IDS research for ICS is summarized in Table 4 while the work related to multiple classes is summarized in Table 5 and 6

7.1 Supervised Learning Approaches

Certain behavior-based approaches have used conventional statistical techniques [97, 205]. These approaches use traditional statistical techniques such as mean and standard deviation on sensor measurements. These techniques are not completely automatic due to their parametric nature. It is difficult to produce statistical tests for a deeply interdependent and large number of sensors and actuators as doing so may lead to unacceptable FPs. ML and DL are considered as non-parametric approaches. They are more automatable and diverse in terms of different techniques employed while using them. In this survey we have grouped the ML approaches as discriminative, generative, and tree-based, with details of each given below.

Table 4: Summary of OCC-based Intrusion Detection work in ICS using Supervised Learning techniques

Work	Domain	Audit Material	Complexity	Algorithms	Feature Selection	Time Series	Dataset	Data Type	Data Available	Metrics
[90]	Conveyor Belt System	Physical	Simple	k-NN, and NB	Yes	Yes	Annon	Actual	No	Confidence, Accuracy
[91]	Chemical Plant	Physical	Simple	OCSVM	Yes	Yes	HITL	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[195]	Annon	Physical	Simple	OCSVM	Yes	Annon	Annon	Actual	No	FPR, FNR
[135]	Gas, and Water	Physical	Simple	SVDD, and KPCA	No	Yes	MSU, and UCI	Actual	Yes	Accuracy
[217]	Water	Physical	Simple	LSTM	Yes	Yes	SWaT	Actual	Yes	Accuracy
[102]	Industrial Demonstra-tor	Physical	Simple	OCSVM, DINA	No	Yes	Industrial demonstrator, and Wind Turbines	Actual, and Simulated	No	TPR, TNR, F1 Score, and Balanced Accuracy
[39]	Water, Gas, and Energy	Nerwork	Simple	ESNN, SOCCADF, OCC-SVM, OCC-CD/CPE	No	Yes	Water, Gas, and Electric	Actual	Yes	TPR, TNR, TA, Precision, Recall, and F1-score
[210]	Annon	Network	Hybrid	SVM	No	Yes	Annon	Actual	No	DR, and IR
[194]	Energy	Physical	Hybrid	DAE, OCSVM, AdaBoost + C4.5, XGBoost, MLP, SVM, k-NN.	Yes	Yes	Annon	Simulated	No	Accuracy, Precision, Recall, and F1 score
[48]	Energy	Physical	Simple	GDLM, SVM, MLP, and PCA	No	Yes	Annon	Simulated	No	F1 Score

7.1.1 Discriminative Approaches. Support Vector Machines (SVM) are linear classifiers, non-probabilistic, and perform binary clas-sification. When using SVM, the data points are projected to a higher dimensional feature space. Then, a hyperplane is learned to distinguish the data points of the two classes. The goal of learn-ing a hyperplane is to enlarge the difference between the closest data points of the classes and thereby provide stronger generaliza-tion on the unseen data. This property of SVM makes it robust for classification problems including IDS [5].

Artificial Neural Networks (ANN) is a class of algorithms that attempt to mimic the learning process of biological neural networks. ANNs are capable of estimating the functions that are dependent on a large number of inputs. There are multiple layers in this network including input, output, and one or more hidden layers. It trains the model to learn the non-linear decision boundaries to segregate the classes. ANNs have also been used for IDS [10].

Instance-based learning algorithms (IBK) do not work on gener-alization as compared to SVM and ANN. Instead, they compute the distance of every new instance with all the available instances in the training dataset. A decision is taken based on all the computed distances. That is why IBK is also referred to as a lazy learning al-gorithm. It has been used in [94, 133, 147] for IDS. The Non-Nested Generalized Exemplars (NNGE) also belong to this class of algo-rithms [149]. It was applied to detect network intrusions in KDDCup 1999 dataset [22].

Artificial immune systems try to mimic the complex vertebrate immune system [213]. They are intelligent and robust computing systems. Fuzzy rules were developed in [199] to express the normal behavior of the system. This was done using Fuzzy-Neural Data Fusion Engine (FN-DFE). Later these fuzzy rules were used for anomaly detection by comparing it with previously described rules of the system. Moreover, a classifier based on neural networks was used to make the concluding decision based on these anomalies.

Multinomial Logistic Regression (LR) is comparable to linear regression and serves as an alternative to Linear Discriminant Anal-ysis (LDA). However, they both have different underlying assump-tions. LR assumes Bernoulli distribution while linear regression assumes Gaussian distribution. Moreover, LR uses the logistic func-tion for prediction. The so predicted values are the probabilities calculated using the logistic function and measure the relationship between the dependent and the independent variable(s). Here, the dependent variable is categorical. Its performance can be improved by using a large number of features. However, it is not as successful in IDS [188].

7.1.2 Decision Tree-Based Approaches. This class also belongs to the discriminative-based approaches but are classified separately due to the existence of distinctive features. This class has been popular among ML researchers. The decisions in this class can be easily translated into an IF-ELSE structure using logical connectives like OR, AND, etc. These decisions (rules) are impulsive and easy to understand. These decisions follow a tree-like structure having

Table 5: Summary of Multiclass-based Intrusion Detection in CPS using Supervised Learning technique (1 of 2)

Work	Domain	Audit Material	Complexity	Algorithms	Feature Selection	Time Series	Dataset	Data Type	Data Available	Metrics
[196]	Energy	Physical	Simple	MSA, SVM, and ANN	No	Yes	PMU	Actual, and Simulated	No	Accuracy
[103]	Healthcare	Physical	Simple	k-NN, NN, SVM, DT, NB, and ZeroR.	No	Yes	Annon	Actual, and Simulated	No	Accuracy, Precision, Recall,and F1 Score
[41]	Water	Network, and Physical	Hybrid	SVM, and SMC	Yes	Yes	Annon	Actual	No	Accuracy, Sensitivity, and Specificity
[16]	Smart Home	Network	Simple	NB, BN, J48, Zero R, OneR, Logistic, SVM, MLP, and RF	Yes	Yes	Annon	Actual	No	F1 Score
[177]	Energy	Physical	SImple	BR with ARD	No	Yes	Annon	Simulated	No	FP, FN, and PT
[203]	Gas, and Energy	Physical	Simple	ELM	Yes	Yes	Annon	Actual, and Simulated	No	ROC, TPR, and FPR
[6]	Water	Network, and Physical	Simple	SVM	Yes	Yes	SWaT, and WADI	Actual	Yes	Accuracy
[14]	Annon	Physical	SImple	CNN	Yes	Yes	Annon	Actual	No	Accuracy
[56]	Water	Network, and Physical	Simple	RF, NBTree, LMT, J48, PART, MLP, HTree, LogF, and SVM.	Yes	Yes	SWaT	Actual	Yes	Precision, and Sensitivity
[176]	Water	Physical	Simple	NN	Yes	Yes	SWaT	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[87]	Electric Vehicles	Network, and Physical	Hybrid	RF, and k-NN	No	Yes	Annon	SImulated	No	Accuracy, DR, ROC, and AUC
[17]	Annon	Physical	Hybrid	LSTM, NN, SVC, and SVM	Yes	Yes	Annon	SImulated	No	Probability of detection
[171]	Annon	Network	Simple	ANN	No	Yes	Annon	Actual	No	Accuracy, Precision, Sensitivity, and ROC
[95]	Cloud	Physical	Simple	LR, RF, NB, RT, SMO, and J48	Yes	Yes	Annon	Actual	No	TPR, TNR, F1 score,and Accuracy
[152]	Energy	Network	Simple	SVM	No	Yes	Annon	Actual	No	Accuracy, and AUC
[46]	Drones	Physical	Simple	GA, XGBoost, and SVM	No	Yes	Annon	Actual	No	Precision
[182]	Energy	Physical	Simple	SVM, k-NN, RF, and CNN	Yes	Yes	Annon	Actual	No	Accuracy
[88]	Cloud	Network	Simple	ELM	Yes	Yes	CTU	Actual	Yes	TPR, FPR, TNR, FNR, Precision, Accuracy, ER, F1 score, MC
[169]	VANETs	Network	Simple	SVM	Yes	Yes	Annon	SImulated	No	DE, FPR, DT and CH Load
[74]	Annon	Network	Simple	AE, LSTM, MLP, SVM, LDA and QDA	Yes	Yes	NSL-KDD	Actual	Yes	Precision, Recall, F1 score, and Accuracy
[159]	Annon	Nework	Simple	BN, NB, MLPNN, J48, and SVM	Yes	Yes	NSL-KDD CUP, and UNSW-NB15	Actual	Yes	DR, FAR, and Accuracy

Table 6: Summary of Multiclass-based Intrusion Detection in CPS using Supervised Learning technique (2 of 2)

Work	Domain	Audit Material	Complexity	Algorithms	Feature Selection	Time Series	Dataset	Data Type	Data Available	Metrics
[57]	Annon	Network	Simple	MLP	Yes	Yes	KDD Cup 99, NSL-KDD, SCX2012, and UNSW-NB15	Actual	Yes	DR, FAR, and AR
[67]	Energy	Physical	Simple	RF, OneR, JRip, Adaboost + JRip,SVM, and NN	Yes	Yes	MSU, and ORNL	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[105]	Supercomputer/ Water	Physical	Simple	LSTM	No	Yes	Tianhe-1A	Actual	Yes	RMSE, and Accuracy
[160]	Healthcare	Physical	Simple	MLP, and SVM	Yes	No	ECG-ID	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[170]	Annon	Network	Simple	MLP, MGSA, PSO, and EBP	No	Yes	Intrusion Detection dataset	Actual	Yes	CCR, ER, MR, and FAR
[145]	Annon	Network	Simple	ASCH-IDS, and RBC-IDS	Yes	No	KDD'99 Dataset	Simulated	Yes	AR, FNR, DR, ROC , and F1 score
[209]	Energy	Network, and Physical	Hybrid	BPNN, and ELM	Yes	Yes	Annon	SImulated	No	Error / Hz
[112]	Vehicle	Network, and Physical	SImple	RNN, MLP, LR, DT(5.0), RF, and SVM	Yes	Yes	Annon	Actual	No	Accuracy
[147]	Annon	Network	Hybrid	k-means-SOM	No	No	KDD-Cup1999	Actual	Yes	FPR, TPR, and DR
[214]	Energy	Network	Simple	SVM, and AIS	No	No	KDD-Cup1999	Simulated	Yes	FPR, FNR, and No. of Detections
[98]	Energy	Physical	Hybrid	Ensemble of SVMs	No	Yes	Bonneville Power Administration	Actual	No	Recall, Precision, F1 score, and Latency
[148]	Energy	Physical	Hybrid	CPM	No	No	MSU Power	Simulated	Yes	Accuracy, and FPR
[28]	Energy	Physical	Simple	NB, OneR, Nnge, Jripper, RF, SVM, and Adaboost	No	No	MSU Power	Simulated	Yes	F1 score
[199]	Energy	Physical	Hybrid	Fuzzy-Neural Data Fusion Engine	No	No	Idaho National Labs energy sys. model	Actual	No	Error Graphs
[23]	Gas	Hybrid	Simple	NB, OneR, Nnge, RF, SVM, and J48	No	No	MSU	Actual	Yes	Precision, and Recall
[52]	Water	Hybrid	Simple	NN	No	No	MSU	Actual	No	Accuracy, FP, and FN
[82]	Water	Physical	Simple	RF, SVM, NN, J48, BN, NB, BFTree, BayesLR, LR, and IBK	No	No	SWaT	Actual	No	Accuracy, AUC, Precision, Recall, and F1 score
[108]	Water	Physical	SImple	RTI+, and BN	Yes	Yes	SWaT	Actual	Yes	CP, and PS

nodes from the top (root) to bottom (leaves). Here the internal nodes can be considered as a test on a feature (attribute). The branches represent the result of the test while leaves represent the labels of the class. Every new record is assigned a label by traversing the tree from the top to the bottom. The selection of attributes as different nodes of the tree is determined based on the information provided by that attribute. In ID3 and J48, this information is calculated through information gain [157, 208]. Information gain is the anticipated reduction in entropy by segregating the examples of datasets based on an attribute. Overfitting can be avoided by proper pruning of the tree. The traversal order of tree is an important factor in this class of algorithm. For example, J48 and Best First Tree (BFTree) are similar to each other. However, BFTree prefers the best node rather than depth-first order. This is a useful approach to prune the trees for avoiding overfitting. One Rule (OneR) is another algorithm of this class. It has one rule for each predictor of the class. The rule with smaller error is selected as "One Rule".

Random Forest (RF) follows an ensemble learning approach [30]. It trains multiple decision trees based on the random subset of features. The majority vote from different decision trees for an instance is selected as the class of that instance. Due to the random selection of features, RF shows different accuracies in every iteration even for the same set of parameters. It is robust in terms of overfitting as compared to the other decision trees. The decision tree algorithms have enjoyed success in IDS at network level [69, 164]. An ensemble method (AdaBoost) was used in [132] for intrusion detection in the network traffic of IoT devices. While IoT devices are playing a vital role in providing comfort to daily routine tasks, they generally have a weak security mechanism. The proposed study used a hybrid approach by using multiple classifiers to detect anomalies. It used Decision Tree, Naive Bayes, and Artificial Neural Network for this purpose. Although the performance of the model is acceptable it suffered from false positives. Also, the ensemble method has more processing time than Decision Tree, Naive Bayes, and Artificial Neural Networks.

7.1.3 Generative Approaches. Generative approaches include Bayesian Classifiers, also referred to as probabilistic classifiers. They predict the class based on the probabilities of any object belonging to a certain class. Bayesian Networks (BayesNet) and Naive Bayes (NB) are two popular Bayesian classifiers used in IDS [85, 201]. The attributes in the NB classifier do not affect each other given the value of the class. They are scalable and the parameter requirement is linear in terms of the number of features. It is suitable for high dimensional data with good generalization over the unseen data. The model is learned in a single iteration over the train data.

BayesNet are directed acyclic graph [49] that represent the set of random variables along with their conditional dependencies. The dependency between the variables can be eliminated by connecting them by an edge. In the real-world, the attributes of a dataset are likely dependent on each other thus making BayesNet approach better than NB, therefore the BayesNet approach is better than NB on a small number of features. BayesLR is a Bayesian variant of LR. It uses Laplace prior to escape from overfitting, and hence is more robust than LR for large feature space [55].

Due to their simplicity, performance, and low computational complexity, Bayesian classifiers are commonly used to solve real-world

problems. A study reported in [108] used the Bayesian networks and RTI+ (Radio Tomographic Imaging) to model the normal behavior of a system and for anomaly detection. Naive Bayes, together with several ML algorithms, was used in [95] to protect the hypervisor or the monitor of a virtual machine. The proposed architecture is composed of executable file extractor, online malware detector, and offline malware classifier. Offline malware classification was accomplished using ML algorithms applied to benign and malicious data.

7.2 Deep Learning Based Supervised Learning Approaches

Deep learning is an extension of ML which focuses on the Artificial Neural Network (ANN). It does not require a complex set of features to be manually engineered by humans, instead they aim to learn these features themselves. This makes them a promising approach for ICS as each ICS has different physical dynamics. Moreover, deep learning is capable of dealing with high-velocity data [86], thus making it desirable for ICS. However, computational complexities associated with deep learning [36] make it difficult.

7.2.1 Convolutional Neural Networks. Convolutional Neural Network (CNN) is a type of deep neural network. It normally works on visual images. CNN is a variant of Multi-Layer Perceptron (MLP). One of the important properties of MLP is Fully Connectedness, in which every single neuron in one layer has connectivity with all neurons in the next layer. This may create the problem of overfitting. To reduce this overfitting, regularization methods are applied. Regularization methods include adjustments of weights to configure the loss function. It exploits the underlying hierarchical pattern of data to get complex patterns from relatively small and simple patterns. CNN has been used in a classification model for different PLC programs using phasor measurement unit (PMU) data generated during the execution of different PLC programs[182]. Later it was used for the anomaly detection in the PMU data. CNN was used to detect keystroke using sensor data of nearby mobile phone[59]. They classified keystrokes using a real-world dataset of 20 users. CNN was used for anomaly detection using thermal side channels [14]. Thermal images were captured on a predefine time window then these images were input in CNN to detect anomalies using the information of predefined actual active time.

7.2.2 Recurrent and Recursive Neural Networks. Recurrent Neural Networks (RNN) is a class of ANN. Its edges input the next time step instead of the next layer of the current time step [100]. RNNs refer to two classes of networks, namely, finite impulse and infinite impulse networks. Both classes have temporal dynamic behavior [125] and could have additional stored states where storage would be controlled by the neural network. Recursive Neural Networks are also a type of deep neural network. They apply the same set of weights recursively over a structured input sequence. Therefore, they give structured predictions over variable-sized input sequences. Compared to RNN, Recursive Neural Networks work hierarchically on the input sequence [100].

Study reported in [112] makes use of RNN to protect vehicles from cyber-attacks. All computations were performed on the cloud. Long short-term memory (LSTM) networks are a type of RNN.

They are effective in speech recognition and showed remarkable performance in speech applications [47]. Though machine learning is being used for intrusion detection, at the same time Adversarial machine learning is being used to counter it. For example, in [217] LSTM was used to train the model on the normal data from a water treatment plant and its performance tested on attack data. Further, the adaptive attacks were performed to deteriorate the performance of the classifier.

State of the art classifiers, including LSTM, were applied on the NSL-KDD dataset to classify the data into different classes for an IDS [74]. A layered architecture using different ML techniques for proactive fault management by predicting sensor values at different stages was proposed in [17]. A hybrid machine learning approach was used to detect anomalies in a simulated IoT environment. Four algorithms, including LSTM, single-layer neural network, SVC, and SVM were used in the anomaly detector. A three-stage layered architecture was proposed for this purpose and served as the quorum for the final decision of the model.

Accurate prediction of faults in supercomputers can be used to overcome financial losses. The historical chilled water data was used in [105] to predict the load of a supercomputer using LSTM. Later, a Z-score model of predicted values was used to identify the anomalies. 5G Networks has proposed a formidable challenge to the security of data, although several anomaly detection mechanisms exist but the emergence of 5G Networks has posed a significant threat due to its high velocity and veracity of data. Deep learning methods were used in [118] for anomaly detection in 5G networks; this was done hierarchically. At the initial level, Deep Belief Network (DBN), or a Stacked AutoEncoder (SAE), was selected to detect the anomaly. At this level, the primary intention was to classify the anomalous data at the high velocity to cope with higher velocity data of 5G Networks and therefore accuracy was not the major concern in this phase. In the subsequent phase, the output of DBN was used by LSTM to recognize the temporal patterns of cyber-attacks.

8 UNSUPERVISED LEARNING

Unsupervised Learning (UL) uses features from feature vector 'X', but there is no corresponding class variable as described in Figure 4. Two types of UL techniques are popular, namely, clustering and Association Rule Mining (ARM). In clustering, different clusters are formed based on a set of feature values while in ARM, rules are extracted based on the support and confidence. The overall work available in literature using UL is summarized in table 7.

8.1 Unsupervised Learning Approaches

8.1.1 Connectivity-based clustering. Connectivity-based clustering is also known as hierarchical clustering, i.e. a hierarchy of clusters is formed. The method can be divided into the following two categories [162]: agglomerative clustering and divisive clustering. Agglomerative clustering proceeds in a bottom-up manner. Here different instances form a cluster with the nearest one at each level of the hierarchy and ultimately forms a single cluster at the top. Divisive clustering proceeds in a top-down manner. In the beginning, a single large cluster is formed which is subsequently divided into smaller clusters at each level of the hierarchy.

8.1.2 Centroid-based clustering. Centroid-based clustering is the most commonly used clustering technique. It works on the concept of the central vector. This central vector does not need to be a member of the dataset. Different clusters are formed based on the central vector. Each instance from the dataset is assigned to each cluster based on its distance to that cluster. K-means is a commonly use centroid-based clustering technique, where k is the fixed number of clusters formed. K-means clustering was used in [27] for the classification of compromised meters in the Advanced Metering Infrastructure (AMI). AMIs are highly vulnerable to false data injection attacks and can be compromised by adversaries to send false data regarding power consumption. In addition to electricity theft, such attacks may also affect the load balancing and other critical functions in a power grid. A consensus correction scheme was introduced in [27] to detect anomaly using the ratio of harmonic to the arithmetic mean. Compromised meter classification was done using k-means clustering. The GRYPHON model is proposed in [39] for anomaly detection in critical infrastructure using evolving spiking neural networks, fuzzy logic, and clustering techniques. It uses fuzzy c-means clustering by assigning random values to cluster centers and subsequently assigns data points to all clusters using the Euclidean distance.

To overcome the vulnerabilities of PLCs, a mechanism to augment PLCs with AES - 256 Encryption and Decryption was proposed in [13]. Further, k-means clustering and Local Outlier Factor (LOF) was used to propose an ML-based intrusion prevention system against three categories of cyber-attacks including interception, injection, and denial of Service. A study reported in [111] used the Channel State Information (CSI) to identify the malicious user in the network. For this purpose, k-means clustering was used to differentiate malicious and legitimate users. Further, this information was used to create an Attack Resilient Profile Builder and Profile Matching Authenticator. Profile Matching was done using SVM.

8.1.3 Distribution-based clustering. Distribution-based clustering is a statistical technique. It works on the principle that if objects belong to the same distribution, then they must be assigned to the same clusters. The technique usually suffers from overfitting unless constraints are applied to the complexity of the model. Gaussian mixture model was applied in [48] to detect false data injection attacks. A mixture Gaussian distribution (MGD) was used to learn the model over normal data.Based on the parameters of this distribution, any upcoming transaction is classified as normal or anomalous. In addition, Principal Component Analysis (PCA), which is an unsupervised machine learning technique, was used for dimensionality reduction. The performance of proposed method was compared with one-class classification (OCC) by using only the normal data. OCC creates a decision boundary on the normal data so that any new transaction on the dataset could be detected whether it is an anomaly or not. The proposed method was also compared with Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP). Overall, the study reported has a good F1 score. The proposed approach has better time complexity than when using SVM and MLP while lower than OCC on training data. It performed better than all of the aforementioned approaches on test data.

8.1.4 Density-based clustering. Density-based clustering works on the principle that higher density data areas need to be separated
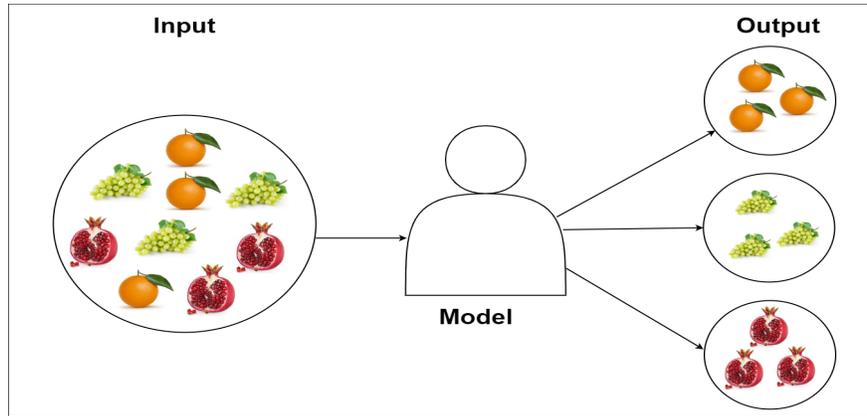
Figure 4: Unsupervised Learning

from the rest of the data. Doing so helps in removing noise and in the creation of a decision boundary. Density-based spatial clustering of applications with noise (DBSCAN) is a well-known density-based clustering technique [42]. It works on the principle of "Density-reachability" using a distance threshold. DBSCAN was used in [32] for anomaly detection in temperature data. Its performance was compared against statistical approaches and several advantages observed in anomaly detection. Likewise, DBSCAN-OD, a variant of DBSCAN for outlier detection, was proposed in [1] for applications with noise. It was able to detect outliers with an accuracy of 99% in simulations.

8.1.5 Grid-based clustering. Grid-based clustering is used in multidimensional datasets [3]. A grid structure is created in this technique and clusters are formed by traversing each cell in the grid based on the threshold density. Grid-based clustering was used in [215] for anomaly detection. They evaluated the system using the Kyoto2006+ and the KDD Cup 1999 datasets. False Positive rate of the proposed algorithm was better than the Song based K-means [179], Song based One-Class SVM [180], Y-means [64], k-means [116], and Li [101]. To partition high dimensional and large data space, a grid-based algorithm was proposed in [197]. The algorithm works in two phases. Firstly, it creates the non-overlapping d-dimensional cells using the domain space followed by partition-based clustering. The proposed approach led to a high detection rate and a relatively low false-positive rate.

8.1.6 Association Rule Mining. ARM [4] is a rule-based machine learning technique used to uncover relationships in databases. Traditionally, it was used for market basket analysis. It has several applications such as predicting customer behavior, product clustering, web usage mining, catalog design, store layout, bioinformatics, and intrusion detection.

ARM works on the principle of Support and Confidence. Support is calculated using the itemset. An itemset is a set of values of one or more attributes. Itemsets that meet the support threshold are called as frequent itemsets. Support for an item set A in D can be defined as the proportion of examples (rows, or transactions) e in

the dataset that contains A. Formally, it can be defined as follows:

$$S(A) = \frac{|e \in D; A \in e|}{|D|} \tag{1}$$

Confidence is the proportion of rules that contain both the antecedent and the consequent. It measures the frequency of the rule w.r.t. the antecedent. The confidence of $X \implies Y$ can be defined as follows:

$$C(X \implies Y) = \frac{S(X \cup Y)}{S(X)}. \tag{2}$$

Frequent itemsets are partitioned in one or more ways to generate rule such as $X \implies Y$, where X is antecedent, and Y the consequent. Rules that satisfy the confidence threshold are qualified for the final set of association rules.

ARM was used in [84] to determine the critical system state for the intrusion detection system using the Apriori algorithm. At the same time, it also incorporated the expert opinion for the identification of critical states. The expert opinion was used in each iteration to reduce the number of candidates in the following iteration. ARM was also used in [146] to generate invariants for a water treatment plant using the Apriori algorithm. This was a preliminary work to discuss the effectiveness of ARM as a proof of concept. It only mined the rules, or invariants, for pairwise sensors/actuators. Secondly, the accuracy of the proposed approach was not effective for practical implementation due to False Positives and False Negatives. Subsequently in [8, 190, 191] invariants were mined on the same plant using the FP-Growth algorithm. The approach succeeded in mining a more exhaustive set of invariants including local and global invariants. Here, "local" refers to within a process and "global" to inter-process invariants. The invariants mined are available at [77]. The invariants were also placed a monitors for distributed attack detection in the plant. The accuracy of the proposed approach was promising considering that the implementation was on an operational plant.

8.2 Deep Learning Based Unsupervised Learning Approaches

There exist several unsupervised deep learning approaches though only a few studies have been reported for securing ICS. Events

Table 7: Summary of Intrusion Detection in CPS using Unsupervised Learning techniques

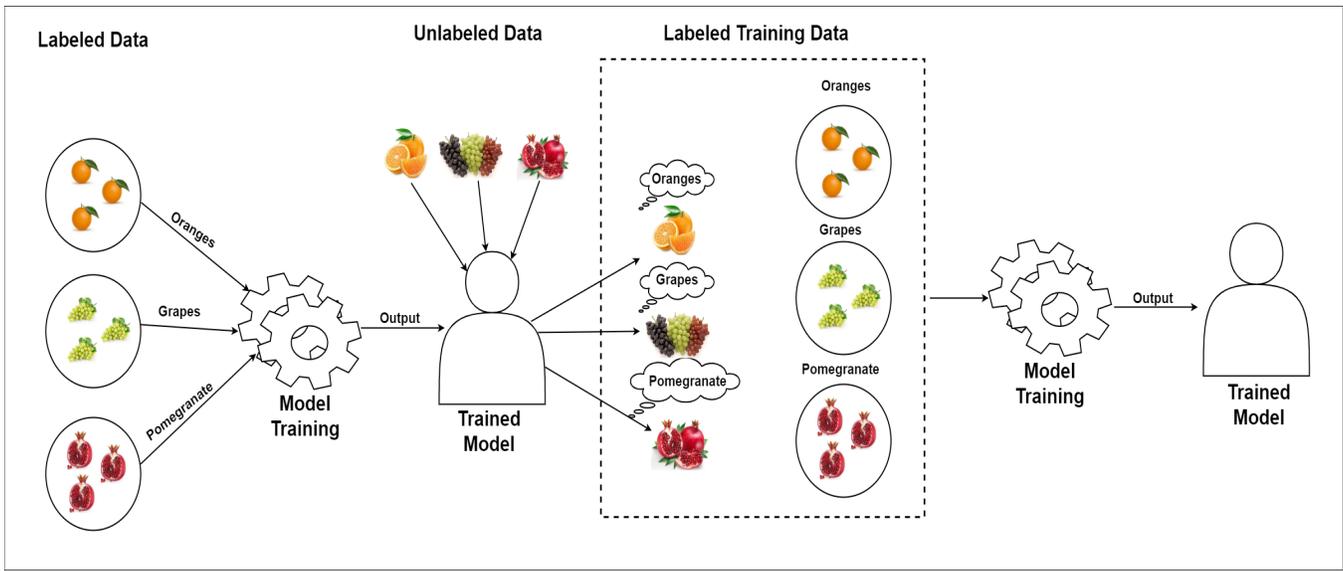| Work	Domain	Audit Material	Complexity	Algorithms	Feature Selection	Time Series	Dataset	Data Type	Data Available	Metrics
[111]	Annon	Network	Simple	k-means, and SVM	No	Yes	Annon	Actual	No	ADR, and Accuracy
[146]	Water	Physical	Simple	Apriori	No	Yes	SWaT	Actual	Yes	Accuracy
[135, 136]	Gas, and Water	Physical	Simple	OCSVM	No	No	MSU	Actual	Yes	Accuracy
[92]	HVAC	Hybrid	Simple	BN	No	No	Annon	Actual	No	Accuracy
[119, 120]	Printed Intelligence	Network	Hybrid	SOM	No	No	PrintoCent	Actual	No	None
[9]	Energy	Physical	Simple	IF, PCA, SVM,k-NN, NB, and MLP	No	Annon	SE-MF	Actual	No	Accuracy, and F1 score
[13]	Water	Network	Simple	k-means, and LOF	Yes	Yes	Annon	Actual	No	PLC Scan Time
[151]	Annon	Network, and Physical	Hybrid	k-NN, SVM, SVR, and AR	Yes	Yes	Annon	Actual	No	SR, and NFAR
[109]	Water	Network	Simple	NN	Yes	Yes	Annon	Actual	No	Recall, and FPR
[84]	Water	Physical	Simple	Apriori	No	No	Annon	Actual	No	Accuracy
[66]	Annon	Network	Simple	PAYL, POSEIDON, Anagram, McPAD	No	No	Annon	Actual	No	FPR, and DR
[173]	Annon	Network	Simple	Multi Hop Clustering	No	No	Annon	Simulated	No	DR
[99]	Aviation, and Robots	Network	Hybrid	Statistical	No	No	Annon	Simulated	No	% of Devient Nodes for convergence
[97]	Energy	Network	Simple	Statistical	No	No	Korean substation	Actual	No	Precision, Recall, F1 score, FPR, and FNR
[165]	Energy	Network	Hybrid	Bayesian	No	Yes	American University of Beirut power plant	Actual	No	Accuracy, and FP
[134]	Gas	Physical	Simple	OCSVM	No	No	MSU	Actual	Yes	Accuracy
[8, 190, 191]	Water	Physical	Simple	FP-growth	Yes	Yes	SWaT	Actual	Yes	Accuracy
[27]	Energy	Physical	Simple	k-means	No	Yes	PeCanStreet Project, and Irish Social Science Data Archive	Actual	Yes	Accuracy
[40]	SCADA	Network	Simple	Statistical	No	No	AUT09	Actual	No	DR, and FP
[205]	SCADA	Hybrid	Simple	Statistical	Yes	Yes	Annon	Simulated	No	Detection Diagrams
[117]	SCADA	Network	Simple	OCSVM	No	No	Annon	Actual	No	Accuracy

Figure 5: Semi-Supervised Learning

originating between the application layer to the kernel layer get recorded in system logs and traces. These logs and traces are helpful in monitoring the performance of any system and are useful for anomaly detection. However, these traces and logs are generally large in a real-time system, and therefore online anomaly detection remains a challenge for such systems. A deep recursive attentive model (DReAM) was proposed in [43] to detect anomalies through temporal information of the system using execution sequences. DReAM works on two components, namely, the unsupervised recurrent neural network predictor and the supervised clustering classifier. Similarly, Mobile Edge Computing (MEC) aims to do intensive computation at the edge networks. This has led to an increase in traffic of transportation networks and the key security issues as well. Therefore, a DL-based framework was proposed in [36] using DBN to learn the model. Its performance was compared with traditional ML-based algorithms. The proposed method was able to detect attacks with acceptable accuracy, but with higher time complexity thus rendering it unsuitable for streaming data. DL-based clustering techniques are further discussed in the following subsections.

8.2.1 Autoencoder based deep clustering. Autoencoder (AE) is a type of ANN that works in an unsupervised manner to learn efficient data encodings [89]. It first learns the representation, i.e., encoding from data and then used for dimensionality reduction. It thus trains the network to ignore noise. It tries to learn a representation close enough to the original input while minimizing the reconstruction loss. There are several AE-based deep clustering methods including Deep Clustering Network (DCN) [204], Deep Embedding Network (DEN) [71], Deep Subspace Clustering Networks (DSC-Nets) [153], Deep Multi-Manifold Clustering (DMC) [33], Deep Embedded Regularized Clustering (DEPICT) [58], and Deep Continuous Clustering (DCC) [167].

8.2.2 Clustering Deep Neural Network (CDNN). This method trains the model primarily on clustering loss. Therefore, if reconstruction loss is not properly designed, then it may lead to a corrupted feature space. Based on network initialization, it can be classified into unsupervised pre-trained, supervised pre-trained, and non-pre-trained network [126].

8.2.3 Variational Autoencoder (VAE)-based deep clustering. In VAE, the latent code of AE is bound to follow a predefined distribution. It is a combination of Bayesian methods [126]. It can use stochastic gradient descent [29] and standard backpropagation [70] to optimize the variational inference.

8.2.4 Generative Adversarial Network (GAN)-based deep clustering. GAN-based clustering works on the principle of the min-max adversarial game. Two types of networks are used, namely, generative and discriminative [126]. The generative network attempts to map a sample from prior distribution to data space whereas the discriminative network maps the input as a real sample of the distribution by computing the probability. There are various GAN-based deep clustering algorithms including Deep Adversarial Clustering (DAC) [68], Categorical Generative Adversarial Network (CatGAN) [181], and Information Maximizing Generative Adversarial Network (InfoGAN) [34].

9 SEMI-SUPERVISED LEARNING

Semi-Supervised Learning (SSL) uses both the labeled and unlabeled data for training of the model, one way of doing SSL is described in Figure 5. In the first phase, the model is trained using the labeled data as in supervised learning. In the second phase, it assigns the labels to the unlabeled data using the model trained in the earlier phase. In the third phase, both the initially given labeled data and the newly assigned labeled data are used for training the model. The following assumptions are made to label the unlabeled data.
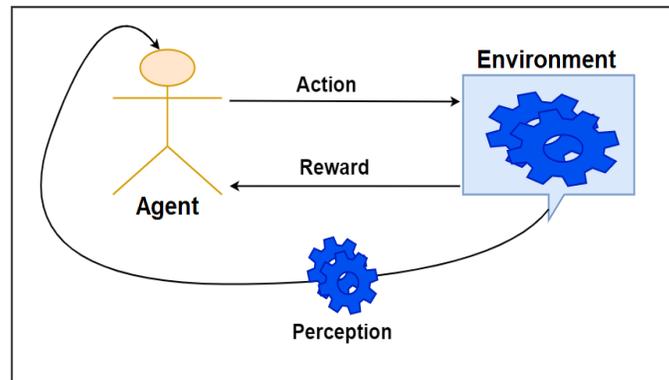
Figure 6: Reinforcement Learning

9.1 Continuity assumption

This assumption works on the principle that points closer to each other are likely to share the same label. This assumption is also used in SL to create decision boundaries. In SSL, this assumption prefers decision boundaries that are in lower density regions. Thus, it is possible that some points are close to each other but may lie in different classes.

9.2 Cluster assumption

This assumption considers that data points are scattered across clusters. The data points present in the same cluster should share the same label.

9.3 Manifold assumption

In this assumption data points lie on a manifold of a lower dimension as compared to the input space. This assumption can eliminate the curse of dimensionality if the manifold is learned using both labeled and unlabeled data. Further learning can be done using distances and densities set out on manifold.

SSL approaches are worth exploring. They have exhibited performance better than supervised and unsupervised approaches when the size of labeled data is relatively small [38, 81, 114]. A study reported in [72] proposed a model to extract the behavioral patterns of malware using semi-supervised and unsupervised machine learning techniques. SSL was used in [73] to automatically update the attack detection system of CPS using the unlabeled malware data. At the first stage, it captures malware patterns from unlabeled data using UL. Then, this information is used by the classification system of the detection engine. The proposed approach used the k-means for clustering and SVM for classification. SSL is also used for fault detection as in [212] using Local Linear Embedding (LLE). LLE is usually used for fault detection in ICS. It only preserves the local information of the structure while ignores the global properties of data. The proposed approach integrated SSL into LLE to utilize the labeled data. The studies reported in [51, 186, 193] have shown that semi-supervised approaches can perform better than supervised and unsupervised approaches for conventional network intrusion detection, but these studies are yet to make their way through to IDS for ICS.

10 REINFORCEMENT LEARNING

Reinforcement Learning (RL) is significantly different from other ML techniques. In RL there exist three main components of the learning system, namely, an Agent, the Environment, and Reward. As illustrated in Figure 6, an agent performs an action in the environment for which it receives reward, which could be positive or negative. This way the agent learns in the environment. RL does not require a dataset for learning as required in other ML techniques. Some commonly used RL algorithms are introduced next.

10.1 Temporal difference (TD) learning

TD is a model-free learning algorithm. The model is learned using bootstrapping done using the current estimate of the value function. Methods are sampled from the environment and updates are performed based on the current estimate [185].

10.2 State–action–reward–state–action (SARSA)

SARSA learns using a Markov Decision Process. It performs actions on the environment and updates its policy based on the reward received against those actions. Initial conditions, learning rate, and discount factor are the hyper-parameters of the algorithm.

10.3 Q-learning

Q-learning is also a model-free algorithm and does not require a model of the environment. It lets the agent learn a policy to perform an action based on different circumstances. It does not require adaptations because of stochastic transitions and rewards.

RL was used in [144] for intrusion detection in a simulated Wireless Sensor Network (WSN) environment. The authors also compared their work with adaptive ML-based IDS. RL-based IDS performed better than other ML-based IDS. A model-free based RL approach was proposed in [96] for anomaly detection in the smart grid. They proposed an RL based solution to the Partially Observable Markov Decision Process (POMDP) problem. For the optimal defense of CPS in [45], the problem was formulated as a two-player zero-sum game. Deep RL was used to tune the actor-critic Neural Network structure. Likewise in [150], a multi-agent general sum game was used to model the attack problem. RL was used to find

the optimal solution for prevention actions and the associated costs. A proof-of-concept was provided by simulating a subsystem of the ATENA controller [18]. Q-Learning based vulnerability assessment of smart grid is reported in [202] where sequential topological attacks were the targets. Using Q-Learning, an attacker can cause severe damage to a plant. The effectiveness of the proposed approach was demonstrated using IEEE 5-bus, RTS-79, and IEEE 300-bus systems-based simulation results. RL was also used in [113] for anomaly detection in Unmanned Aerial Vehicle (UAV). It recorded the temperature of the motor using sensors and used a Raspberry Pi based processing unit to observe the anomalous behavior of the motor.

11 MAJOR CHALLENGES AND RECOMMENDATIONS FOR IDS IN ICS

11.1 Adversarial Machine Learning for IDS

Machine learning is being used for intrusion detection, at the same time Adversarial machine learning is being used to counter its benefits. For example, in [217] LSTM was used to train the model on the normal data from a real-world ICS and its performance tested on attack data. Further, the adaptive attacks were performed to deteriorate the performance of the machine learning classifier. Machine Learning as a service (MLaas) is also gaining popularity in cloud-based services. They typically use deep neural networks (DNN) for different predictive models. Now they have become vulnerable to different adversarial attacks. In this case, the adversary try to steal the model by querying the Application Programming Interface (API). For example a study proposed in [207] used an attack methodology to extract the DNN models from various cloud-based platforms. For this purpose, they used various algorithms including active and transfer learning. Similarly, [107] used composite attacks using Trojan triggers to disrupt the performance of DNN model. Their Trojan triggers were composed of benign features of multiple labels. The model misclassifies the output when input is stamped with Trojan trigger.

There are some studies which have tried to tackle this challenge but still more work needs to be done. For example, [211] used the zero knowledge proofs for decision tree. They reported their accuracy and predictions on public dataset without leaking any information about the model. Using the proposed study a decision tree having depth of 23 levels and 1029 number of nodes can generate the zero knowledge proofs in 250 seconds. Likewise, [104] used simple and smaller pre-trained neural network models for the verification of DNN-based systems and to protect them against adversarial attacks. We believe that these types of approaches could be useful for defending the adversarial attacks on machine learning models.

11.2 Lack of Attack Patterns and its Mitigation in ICS

It is difficult to produce an exhaustive dictionary of attack signature in complex physical processes in ICS. Therefore it becomes difficult to detect zero-day attacks. For example, the model proposed in [137] is robust and fast as it does not require training on new input data.

However, as it generates signatures using only the available malware processes, it could be prone to zero-day attacks. The study reported in [189] used the unsupervised machine learning approach to generate attacks for a real-world ICS. They used association rule mining to generate attack patterns. Normally, Supervised learning approaches lacks attack data, therefore the study reported in [189] could be beneficial for making robust supervised learning-based IDS. Moreover, it can also be useful for signature-based approaches for IDS, as it automatically generates the signatures using the attacked data on real-world ICS. Cyber-attacks were modeled as timed-automaton in [183] for SWaT [121]. This model was used as a baseline to create a number of cyber-attacks using mutation. Though all the created attacks may not be actual attacks but it seems to be a good strategy to defend against zero-day attacks because of the comprehensive attack dictionary created by the proposed approach. Similarly, a study reported in [78] used a gradient-based attack scheme to generate attacks for real-world ICS. Through their approach they mislead the RNN based anomaly detector of two real-world ICS namely SWaT [121] and WADI [7].

11.3 Aging and Complexity of the Physical Systems in ICS

There are serious issues related to the aging and complexity of the physical systems while dealing with specification-based approach. There could be inaccuracies in the operational manuals, and interpretation of the process behavior. Though behavior-based approach is favoured against incorrect vendor specifications due to its dependability on empirical data but there are issues of detecting zero-day attacks, ensuring an acceptable rate of false alarms, and managing computational complexity

11.4 Heterogeneity among Physical Processes in ICS

There exists a heterogeneous behavior among physical processes of an ICS because components, size, and functionality of each process is different from others [77, 121]. Therefore it is a challenge to detect attacks in the heterogeneous physical processes controlled by an ICS. For example, SWaT testbed [77] which is an industrial scaled-down replica of a water treatment plant has different six stages. Here attack on one stage can disrupt the processes of other stages as well. So developing a model which can capture the behavior of heterogeneous physical processes is still a major challenge. Though IDS based on physical process have received relatively little attention but now there is a growing trend to detect intrusion at the physical process level.

11.5 Inherent Class Imbalance Nature of IDS

Behavior-based approaches suffer from skewed class problems. Here most of the data belongs only to a single class (normal behavior). Any naive classifier that labels each instance as normal will get a higher accuracy. Therefore, accuracy is not enough to assess the performance of IDS. It is also important to note that acceptable values of the metrics discussed in section 5.7 might still not make an IDS suitable for deployment in an operational plant. As an example, consider accuracy. A high accuracy can be obtained by having high values of TP and TN and relatively lower values of

FP and FN. However, suppose that accuracy is high, lets say, 99%, but the number of false positives (FP) per day is, say, 50. Such an IDS would likely be not used in an operational plant. Thus, it is recommended that in addition to reporting one or more metrics mentioned above, FP must also be reported to assess how well an IDS would perform when deployed in a constantly running plant.

11.6 Stealthy Attacks on ICS

If the attacker has deep insights of the system then it would be vulnerable to stealthy attacks. These types of attacks gradually disrupt the performance of the operational plant. Though there are a number of studies on this issue but the detection of these attacks is still a major challenge. For example, a study proposed in [184] used the Profile-DNS for detecting the stealthy attacks by characterizing the expected DNS behavior. Likewise, [128] used VMshield for securing the cloud platforms against the stealthy attacks. They did feature selection using meta-heuristic , and binary particle swarm optimization (BPSO) algorithms. They used Random Forest for the classification of malicious and benign processes.

11.7 Association Rule Mining for IDS

Most of the reported ML-based intrusion detection work in ICS uses SL approaches while there exists only a sprinkling of work using UL approaches. Particularly, only a few studies have reported the use of an ARM-based UL approach for intrusion detection in ICS [84, 146, 190, 191]. Despite this, there remain gaps that need to be filled. For example, all the studies reported in [84, 146, 190, 191] used data from an ICS controlling a water plant. Though, [191] practically implemented the ARM-approach in an operational plant with promising results, the same approach needs to be tested on other ICS used in systems such as the smart grid and gas plants. Moreover, [191] used a time series data but used the FP-Growth algorithm to mine the rules. FP-Growth is time agnostic, therefore could be promising to use Temporal Association Rule Mining [106] for that purpose.

11.8 Deep Learning for IDS

Deep learning can be an effective approach for detecting anomalies in ICS-controlled plants. It can automatically generate features based on the physical dynamics of each ICS. Some studies reported the use of DL to secure ICS as discussed in section 7.2 and 8.2 most of which are SL approaches. There exist only a few studies [36, 43] where the UL approach is used. There are several DL-based clustering techniques as discussed in section 8.2 that need to be explored for securing the ICS. However, higher time complexity possesses a great challenge for their application in real-time systems, such as ICS.

11.9 Agent-based Learning for Securing ICS

Reinforcement Learning which works on the basis of agent and environment interaction is the least explored area for securing ICS. Though there are a few studies reported in the literature as discussed in section 10. However, considering the dynamic nature of ICS in different domains including smart grids, water, gas, etc, RL appears a promising avenue to explore and implement in various domains.

11.10 Zero-shot Learning for Resilience against Zero-day Attacks

Apart from the ones mentioned above, there are other promising ML approaches that need to be explored for intrusion detection in ICS. Zero-shot learning [163, 175] is one such promising approach for detecting zero-day attacks. Domain adaptation can help learn an IDS for one ICS using data of some other ICS. Lastly, distribution shift techniques can be explored for making the model adapt to the changing behavior of the system with time. The above-mentioned approaches remain to be explored in depth to effectively solve the problem of intrusion detection.

11.11 Comparative Analysis of Behavior-based Approach with Specification and Signature-based Approach for IDS

Even though some studies compare various ML algorithms on their dataset, we were not able to find a comparison with the specification or signature-based techniques except in [190, 191] where the behavior-based approach was compared with the specification based approach. A comparison of the three types of approaches needs to be conducted on the same dataset under similar assumptions to gain a better understanding of their effectiveness in detecting cyber-attacks.

11.12 Need of Comprehensive Evaluation Metrics for Real-world ICS

Several studies have reported only a few metrics such as either accuracy (or error rate/graphs) or detection rates as discussed in section 5.7. Reporting only one or two performance metrics for a skewed class problem is not sufficient, more than one of the following metrics should be used: accuracy, precision, recall, F-measure, ROC, and AUC. Only a few studies have reported AUC or ROC despite the fact that these are more appropriate measure of classifier performance in IDS. Secondly, there is little focus in the literature to report the time to detect an attack or the percentage detection of an attack over the duration for which it lasts. The use of these measures should be made more prevalent.

11.13 Multi-layered Defense for IDS

A majority of the approaches focus on detecting intrusion at the network layer. After all, this is the first line of defense of an ICS, though often easier to breach as many ICS are using ready-made industrial protocols, and due to insider threats. Once breached, detecting intrusions in the physical layer improves the chances of avoiding plant damage or service disruption. Detecting cyber-attacks at this layer would be more promising as each ICS is unique and to be successful, the attacker would require a knowledge of the physical dynamics of that particular ICS. Therefore more attention seems necessary in developing IDS for the physical layer consisting of at least a few dozen sensors and actuator attributes. We believe that the final solution lies in a multi-layered defense, a network IDS followed by a physical process IDS.

11.14 Root Cause Isolation for IDS

While detecting a cyber-attack launched by an intruder is the primary goal of an IDS, detecting the nature and location of the ongoing attack, and taking further actions, remain crucial to steps. Only one work [92] has reported root cause isolation. Lastly, few studies have modeled the problem as a time series problem, whereas, many ICS repeat the same operations over and over again. More approaches are needed to address these issues.

12 CONCLUSION

ICS are critical for the economy and infrastructure of any country and hence ought to be protected against cyber-adversaries. These adversaries could be hackers, enemy states, and displeased employees, etc. Therefore securing an ICS from cyber-attacks is one of the prime concerns for governments and organizations. Behavior-based approaches such as machine learning, deep learning, and statistical approaches for intrusion detection, are gaining attention. They can be automated, several scale well, and can be generalized and are becoming affordable to apply because of cheaper and widely available computational power. Thus, this survey focuses on literature to consolidate the work on behavior-based approaches for IDS in ICS, categorizes them, identifies gaps, and proposes future research directions. This area This area is in a need of a high fidelity benchmark dataset. There is room to apply the newly developed ML techniques and compare them with the specification and signature-based approaches, especially for the physical process controlled by an ICS. Time series modeling of the problem and the use of new metrics is also required. All in all, ML and DL approaches are promising techniques for the detection of cyber-attacks in both the network and physical process layer of an ICS, though there is room for improvement.

ACKNOWLEDGEMENTS

REFERENCES

[1] Aymen Abid, Abdennaceur Kachouri, and Adel Mahfoudhi. 2017. Outlier detection for wireless sensor networks using density-based clustering approach. *IET Wireless Sensor Systems* 7, 4 (2017), 83–90.
[2] Sridhar Adepu and Aditya Mathur. 2016. Using Process Invariants to Detect Cyber Attacks on a Water Treatment System. In *ICT Systems Security and Privacy Protection*, Jaap-Henk Hoepman and Stefan Katzenbeisser (Eds.). Springer International Publishing, Cham, 91–104.
[3] Charu C. Aggarwal and Chandan K. Reddy. 2013. *Data Clustering: Algorithms and Applications* (1st ed.). Chapman & Hall/CRC.
[4] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. 1993. Mining Association Rules between Sets of Items in Large Databases. *SIGMOD Rec.* 22, 2 (June 1993), 207–216. https://doi.org/10.1145/170036.170072
[5] Iftikhar Ahmad, Muhammad Hussain, Abdullah Alghamdi, and Abdulhameed Alelaiwi. 2014. Enhancing SVM performance in intrusion detection using optimal feature subset selection based on genetic principal components. *Neural Computing and Applications* 24, 7-8 (2014), 1671–1682.
[6] Chuadhry Mujeeb Ahmed, Martin Ochoa, Jianying Zhou, Aditya P. Mathur, Rizwan Qadeer, Carlos Murguia, and Justin Ruths. 2018. <i>NoisePrint</i>: Attack Detection Using Sensor and Process Noise Fingerprint in Cyber Physical Systems. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security* (Incheon, Republic of Korea) *(ASIACCS '18)*. Association for Computing Machinery, New York, NY, USA, 483–497. https://doi.org/10.1145/3196494.3196532
[7] Chuadhry Mujeeb Ahmed, Venkata Reddy Palleti, and Aditya P. Mathur. 2017. WADI: A Water Distribution Testbed for Research in the Design of Secure Cyber Physical Systems. In *CysWater*. ACM, NY, USA. http://doi.acm.org/10.1145/3055366.3055375
[8] Chuadhry Mujeeb Ahmed, Muhammad Azmi Umer, Beebi Siti Salimah Binte Liyakkathali, Muhammad Taha Jilani, and Jianying Zhou. 2021. Machine Learning for CPS Security: Applications, Challenges and Recommendations. In *Machine Intelligence and Big Data Analytics for Cybersecurity Applications*. Springer, 397–421.
[9] Saeed Ahmed, YoungDoo Lee, Seung-Ho Hyun, and Insoo Koo. 2019. Unsupervised Machine Learning-Based Detection of Covert Data Integrity Assault in Smart Grid Networks Utilizing Isolation Forest. *IEEE Transactions on Information Forensics and Security* 14, 10 (2019), 2765–2777.
[10] Omar Al-Jarrah and Ahmad Arafat. 2015. Network Intrusion Detection System Using Neural Network Classification of Attack Behavior. *Journal of Advances in Information Technology Vol* 6, 1 (2015), 1–8.
[11] M. Alabadi and Z. Albayrak. 2020. Q-Learning for Securing Cyber-Physical Systems : A survey. In *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. IEEE, 1–13.
[12] A. M. Aleesa, B. B. Zaidan, A. A. Zaidan, and Nan M. Sahar. 2020. Review of intrusion detection systems based on deep learning techniques: coherent taxonomy, challenges, motivations, recommendations, substantial analysis and future directions. *Neural Comput. Appl.* 32, 14 (2020), 9827–9858. https://doi.org/10.1007/s00521-019-04557-3
[13] Thiago Alves, Rishabh Das, and Thomas Morris. 2018. Embedding encryption and machine learning intrusion prevention systems on programmable logic controllers. *IEEE Embedded Systems Letters* 10, 3 (2018), 99–102.
[14] Hussam Amrouch, Prashanth Krishnamurthy, Naman Patel, Jörg Henkel, Ramesh Karri, and Farshad Khorrami. 2017. Emerging (un-) reliability based security threats and mitigations for embedded systems: Special session. In *Proceedings of the 2017 International Conference on Compilers, Architectures and Synthesis for Embedded Systems Companion*. IEEE, 1–10.
[15] Tiranuch Anantvalee and Jie Wu. 2007. A survey on intrusion detection in mobile ad hoc networks. In *Wireless Network Security*. Springer, 159–180.
[16] Eirini Anthi, Lowri Williams, Małgorzata Słowińska, George Theodorakopoulos, and Pete Burnap. 2019. A Supervised Intrusion Detection System for Smart Home IoT Devices. *IEEE Internet of Things Journal* 6, 5 (2019), 9042–9053.
[17] V Ariharan, Subha P Eswaran, Srinivasarao Vempati, and Naveed Anjum. 2019. Machine Learning Quorum Decider (MLQD) for Large Scale IoT Deployments. *Procedia Computer Science* 151 (2019), 959–964.
[18] AUBIGNY. 2017. A. Consortium, "ATENA website". https://www.atena-h2020.eu/.
[19] Stefan Axelsson. 2000. *Intrusion detection systems: A survey and taxonomy*. Technical Report. Technical report Chalmers University of Technology, Goteborg, Sweden.
[20] Radhakisan Baheti and Helen Gill. 2011. Cyber-physical systems. *The impact of control technology* 12, 1 (2011), 161–166.
[21] José Barbosa, Paulo Leitão, Damien Trentesaux, Armando W Colombo, and Stamatis Karnouskos. 2016. Cross benefits from cyber-physical systems and intelligent products for future smart industries. In *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*. IEEE, 504–509.
[22] Stephen D Bay, Dennis Kibler, Michael J Pazzani, and Padhraic Smyth. 2000. The UCI KDD archive of large data sets for data mining research and experimentation. *ACM SIGKDD Explorations Newsletter* 2, 2 (2000), 81–85.
[23] Justin M Beaver, Raymond C Borges-Hink, and Mark A Buckner. 2013. An evaluation of machine learning methods to detect malicious SCADA communications. In *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, Vol. 2. IEEE, 54–59.
[24] R. Berthier and W. H. Sanders. 2011. Specification-Based Intrusion Detection for Advanced Metering Infrastructures. In *2011 IEEE 17th Pacific Rim International Symposium on Dependable Computing*. 184–193.
[25] Robin Berthier and William H Sanders. 2011. Specification-based intrusion detection for advanced metering infrastructures. In *Dependable Computing (PRDC), 2011 IEEE 17th Pacific Rim International Symposium on*. IEEE, 184–193.
[26] Deval Bhamare, Maede Zolanvari, Aiman Erbad, Raj Jain, Khaled Khan, and Nader Meskin. 2020. Cybersecurity for industrial control systems: A survey. *Computers & Security* 89 (2020), 101677. https://doi.org/10.1016/j.cose.2019.101677
[27] Shameek Bhattacharjee, Aditya Thakur, and Sajal K. Das. 2018. Towards Fast and Semi-Supervised Identification of Smart Meters Launching Data Falsification Attacks. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security* (Incheon, Republic of Korea) *(ASIACCS '18)*. Association for Computing Machinery, New York, NY, USA, 173–185. https://doi.org/10.1145/3196494.3196551
[28] Raymond C Borges Hink, Justin M Beaver, Mark A Buckner, Tommy Morris, Uttam Adhikari, and Shengyi Pan. 2014. Machine learning for power system disturbance and cyber-attack discrimination. In *Resilient Control Systems (ISRCS), 2014 7th International Symposium on*. IEEE, 1–8.
[29] Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*. Springer, 177–186.

Table 8: Abbreviations used in the survey.*Terms in bold are used in machine learning literature.

Term*	Expansion	Term*	Expansion
AE	Autoencoder	**LDA**	Linear Discriminent Analysis
AMI	Advanced Metering Infrastructure	**LLE**	Local Linear Embedding
ANN	Artificial Neural Networks	**LR**	Logistic Regression
ARM	Association Rule Mining	**LSTM**	Long-Short Term Memory
AUC	Area Under ROC	**MLP**	Multi-Layer Perceptron
BACnet	Building Automation Control Network	**NB**	Naive Bayes
BayesLR	Bayes Logistic Regression	**NNGE**	Non-Nested Generalized Exemplers
BayesNet	Bayes Network	**OCC**	One-Class Classification
BFTree	Best First Tree	**OneR**	One Rule
CatGAN	Categorical Generative Adversarial Network	PLC	Programmable Logic Controller
CDNN	Clustering Deep Neural Network	PMU	Phasor Management Unit
CNN	Convolutional Neural Networks	**POMDP**	Partially Observable Markov Decision Process
CPS	Cyber-Physical System	**RL**	Reinforcement Learning
DAC	Deep Adversarial Clustering	**RF**	Random Forest
DBN	Deep Belief Networks Decision Process	**RNN**	Recurrent Neural Networks
DCC	Deep Continuous Clustering	**ROC**	Receiver Operating Characteristic
DEN	Deep Embedding Network	**SAE**	Stacked Autoencoder
DEPICT	Deep Embedded Regularized Clustering	**SARSA**	State–action–reward–state–action
DL	Deep learning	SCADA	Supervisory Control and data Acquisition System
DMC	Deep Multi-Manifold Clustering	**SL**	Supervised Learning
DNP3	Distributed Network Protocol 3	**SOM**	Self-Organizing Maps
DReAM	Deep Recursive Attentive Model	**SVM**	Support Vector Machine
DSC-Nets	Deep Subspace Clustering Networks	**SSL**	Semi-Supervised Learning
FN	False Negative	TCP	Transmission Control Protocol
FP	False Positive	**TD**	Temporal difference
GAN	Generative Adversarial Network	**TP, TPR**	True Positive, True Positive Rate
ICMP	Internet Control Message Protocol	**TN**	True Negative
ICS	Industrial Control Systems	UAV	Unmanned Aerial Vehicle
IDS	Intrusion Detection System	UDP	User Datagram Protocol
InfoGAN	Information Maximizing Generative Adversarial Network	**UL**	Unsupervised Learning
ML	machine learning	**VAE**	Variational Autoencoder
LAN	Local Area Network		

[30] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

[31] Alvaro A. Cárdenas, Saurabh Amin, Zong-Syun Lin, Yu-Lun Huang, Chi-Yen Huang, and Shankar Sastry. 2011. Attacks against Process Control Systems: Risk Assessment, Detection, and Response. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security* (Hong Kong, China) *(ASIACCS '11)*. Association for Computing Machinery, New York, NY, USA, 355–366. https://doi.org/10.1145/1966913.1966959

[32] Mete Çelik, Filiz Dadaşer-Çelik, and Ahmet Şakir Dokuz. 2011. Anomaly detection in temperature data using dbscan algorithm. In *2011 International Symposium on Innovations in Intelligent Systems and Applications*. IEEE, 91–95.

[33] Dongdong Chen, Jiancheng Lv, and Yi Zhang. 2017. Unsupervised multi-manifold clustering by learning deep representation. In *Workshops at the thirty-first AAAI conference on artificial intelligence*. AAAI.

[34] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. In *Proceedings of the 30th International Conference on Neural Information Processing Systems* (Barcelona, Spain) *(NIPS'16)*. Curran Associates Inc., Red Hook, NY, USA, 2180–2188.

[35] You Chen, Yang Li, Xue-Qi Cheng, and Li Guo. 2006. Survey and taxonomy of feature selection algorithms in intrusion detection system. In *Information security and cryptology*. Springer, 153–167.

[36] Yuanfang Chen, Yan Zhang, Sabita Maharjan, Muhammad Alam, and Ting Wu. 2019. Deep learning for secure mobile edge computing in cyber-physical transportation systems. *IEEE Network* 33, 4 (2019), 36–41.

[37] Steven Cheung, Bruno Dutertre, Martin Fong, Ulf Lindqvist, Keith Skinner, and Alfonso Valdes. 2007. Using model-based intrusion detection for SCADA networks. In *Proceedings of the SCADA security scientific symposium*, Vol. 46. Citeseer, 1–12.

[38] Antonio Criminisi, Jamie Shotton, and Ender Konukoglu. 2012. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and Trends® in Computer Graphics and Vision* 7, 2–3 (2012), 81–227.

[39] Konstantinos Demertzis, Lazaros Iliadis, and Ilias Bougoudis. 2019. Gryphon: a semi-supervised anomaly detection system based on one-class evolving spiking neural network. *Neural Computing and Applications* (2019), 1–12.

[40] Patrick Düssel, Christian Gehl, Pavel Laskov, Jens-Uwe Bußer, Christof Störmann, and Jan Kästner. 2009. Cyber-critical infrastructure protection using

real-time payload-based anomaly detection. In *Critical Information Infrastructures Security*. Springer, 85–97.

[41] Ibrahim Elgendi, Md Farhad Hossain, Abbas Jamalipour, and Kumudu S Munasinghe. 2019. Protecting cyber physical systems using a learned MAPE-K model. *IEEE Access* 7 (2019), 90954–90963.

[42] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (Portland, Oregon) *(KDD'96)*. AAAI Press, 226–231.

[43] Okwudili M Ezeme, Qusay H Mahmoud, and Akramul Azim. 2019. DReAM: Deep recursive attentive model for anomaly detection in kernel events. *IEEE Access* 7 (2019), 18860–18870.

[44] Nicolas Falliere, Liam O Murchu, and Eric Chien. 2011. W32. stuxnet dossier. *White paper, Symantec Corp., Security Response* 5 (2011).

[45] Ming Feng and Hao Xu. 2017. Deep reinforcement learning based optimal defense for cyber-physical system in presence of unknown cyber-attack. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 1–8.

[46] Zhiwei Feng, Nan Guan, Mingsong Lv, Wenchen Liu, Qingxu Deng, Xue Liu, and Wang Yi. 2020. Efficient drone hijacking detection using two-step GA-XGBoost. *Journal of Systems Architecture* 103 (2020), 101694.

[47] Santiago Fernández, Alex Graves, and Jürgen Schmidhuber. 2007. An application of recurrent neural networks to discriminative keyword spotting. In *International Conference on Artificial Neural Networks*. Springer, 220–229.

[48] S Armina Foroutan and Farzad R Salmasi. 2017. Detection of false data injection attacks against state estimation in smart grids based on a mixture Gaussian distribution learning method. *IET Cyber-Physical Systems: Theory & Applications* 2, 4 (2017), 161–171.

[49] Nir Friedman, Dan Geiger, and Moises Goldszmidt. 1997. Bayesian network classifiers. *Machine learning* 29, 2-3 (1997), 131–163.

[50] Josh Fruhlinger. 2018. What is WannaCry ransomware, how does it infect, and who was responsible? https://www.csoonline.com/article/3227906/what-is-wannacry-ransomware-how-does-it-infect-and-who-was-responsible.html.

[51] Guohong Gao, Guoyi Miao, Jiaxia Sun, and Yafeng Han. 2013. Improved Semi-supervised Fuzzy Clustering Algorithm and Application in Effective Intrusion Detection System. *International Journal of Advancements in Computing Technology* 5, 4 (2013).

[52] Wei Gao, Thomas Morris, Bradley Reaves, and Drew Richey. 2010. On SCADA control system command and response injection and intrusion detection. In *eCrime Researchers Summit (eCrime), 2010*. IEEE, 1–9.

[53] Wei Gao and Thomas H Morris. 2014. On Cyber Attacks and Signature Based Intrusion Detection for MODBUS Based Industrial Control Systems. *Journal of Digital Forensics, Security and Law* 9, 1 (2014), 37–56.

[54] Pedro Garcia-Teodoro, J Diaz-Verdejo, Gabriel Maciá-Fernández, and Enrique Vázquez. 2009. Anomaly-based network intrusion detection: Techniques, systems and challenges. *computers & security* 28, 1 (2009), 18–28.

[55] Alexander Genkin, David D Lewis, and David Madigan. 2007. Large-scale Bayesian logistic regression for text categorization. *Technometrics* 49, 3 (2007), 291–304.

[56] Hamid Reza Ghaeini, Nils Ole Tippenhauer, and Jianying Zhou. 2019. Zero Residual Attacks on Industrial Control Systems and Stateful Countermeasures. In *Proceedings of the 14th International Conference on Availability, Reliability and Security* (Canterbury, CA, United Kingdom) *(ARES '19)*. Association for Computing Machinery, New York, NY, USA, Article 80, 10 pages. https://doi.org/10.1145/3339252.3340331

[57] Waheed AHM Ghanem and Aman Jantan. 2019. A new approach for intrusion detection system based on training multilayer perceptron by using enhanced Bat algorithm. *Neural Computing and Applications* (2019), 1–34.

[58] Kamran Ghasedi Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, and Heng Huang. 2017. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *Proceedings of the IEEE international conference on computer vision*. IEEE, 5736–5745.

[59] Tyler Giallanza, Travis Siems, Elena Smith, Erik Gabrielsen, Ian Johnson, Mitchell A Thornton, and Eric C Larson. 2019. Keyboard Snooping from Mobile Phone Arrays with Mixed Convolutional and Recurrent Neural Networks. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 1–22.

[60] Jairo Giraldo, David Urbina, Alvaro Cardenas, Junia Valente, Mustafa Faisal, Justin Ruths, Nils Ole Tippenhauer, Henrik Sandberg, and Richard Candell. 2018. A Survey of Physics-Based Attack Detection in Cyber-Physical Systems. *ACM Comput. Surv.* 51, 4, Article 76 (July 2018), 36 pages. https://doi.org/10.1145/3203245

[61] Jonathan Goh, Sridhar Adepu, Khurum Nazir Junejo, and Aditya Mathur. 2016. A dataset to support research in the design of secure water treatment systems. In *International Conference on Critical Information Infrastructures Security*. Springer, 88–99.

[62] GReAT. 2015. The Mystery of Duqu 2.0: a sophisticated cyberespionage actor returns. https://securelist.com/the-mystery-of-duqu-2-0-a-sophisticated-cyberespionage-actor-returns/70504/.

[63] GReAT. 2018. Hades, the actor behind Olympic Destroyer is still alive. https://securelist.com/olympic-destroyer-is-still-alive/86169/.

[64] Y. Guan, A. A. Ghorbani, and N. Belacel. 2003. Y-means: a clustering method for intrusion detection. In *CCECE 2003 - Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology (Cat. No.03CH37436)*, Vol. 2. 1083–1086 vol.2.

[65] Rachana Ashok Gupta and Mo-Yuen Chow. 2010. Networked control system: overview and research trends. *Industrial Electronics, IEEE Transactions on* 57, 7 (2010), 2527–2535.

[66] Dina Hadžiosmanović, Lorenzo Simionato, Damiano Bolzoni, Emmanuele Zambon, and Sandro Etalle. 2012. N-gram against the machine: On the feasibility of the n-gram network analysis for binary protocols. In *Research in Attacks, Intrusions, and Defenses*. Springer, 354–373.

[67] Lida Haghnegahdar and Yong Wang. 2019. A whale optimization algorithm-trained artificial neural network for smart grid cyber intrusion detection. *Neural Computing and Applications* (2019), 1–15.

[68] Warith Harchaoui, Pierre-Alexandre Mattei, and Charles Bouveyron. 2017. Deep adversarial Gaussian mixture auto-encoder for clustering. In *International Conference on Learning Representations*. ICLR, Toulon, France.

[69] Md Al Mehedi Hasan, Mohammed Nasser, Biprodip Pal, and Shamim Ahmad. 2014. Support vector machine and random forest modeling for intrusion detection systems. *Journal of Intelligent Learning Systems and Applications* 2014 (2014), 45–52.

[70] ROBERT HECHT-NIELSEN. 1992. III.3 - Theory of the Backpropagation Neural Network**Based on "nonindent" by Robert Hecht-Nielsen, which appeared in Proceedings of the International Joint Conference on Neural Networks 1, 593–611, June 1989. © 1989 IEEE. In *Neural Networks for Perception*, Harry Wechsler (Ed.). Academic Press, 65 – 93. https://doi.org/10.1016/B978-0-12-741252-8.50010-8

[71] Peihao Huang, Yan Huang, Wei Wang, and Liang Wang. 2014. Deep embedding network for clustering. In *2014 22nd International conference on pattern recognition*. IEEE, 1532–1537.

[72] Shamsul Huda, Jemal Abawajy, Baker Al-Rubaie, Lei Pan, and Mohammad Mehedi Hassan. 2019. Automatic extraction and integration of behavioural indicators of malware for protection of cyber–physical networks. *Future Generation Computer Systems* 101 (2019), 1247–1258.

[73] Shamsul Huda, Suruz Miah, Mohammad Mehedi Hassan, Rafiqul Islam, John Yearwood, Majed Alrubaian, and Ahmad Almogren. 2017. Defending unknown attacks on cyber-physical systems by semi-supervised approach and available unlabeled data. *Information Sciences* 379 (2017), 211–228.

[74] Cosimo Ieracitano, Ahsan Adeel, Francesco Carlo Morabito, and Amir Hussain. 2020. A novel statistical analysis and autoencoder driven intelligent intrusion detection approach. *Neurocomputing* 387 (2020), 51 – 62. https://doi.org/10.1016/j.neucom.2019.11.016

[75] INCIBE. 2019. Aurora vulnerability: origin, explanation and solutions. https://www.incibe-cert.es/en/blog/aurora-vulnerability-origin-explanation-and-solutions.

[76] Jun INOUE, Yoriyuki YAMAGATA, Yufi CHEN, Christopher M. POSKITT, , and Jun SUN. 2017. Anomaly detection for a water treatment system using unsupervised machine learning. In *Proceedings of 17th IEEE International Conference on Data Mining Workshops ICDMW 2017, 18-21 November*. IEEE, New Orleans, LA, 1058–1065.

[77] iTrust. 2015. Dataset and Models. https://itrust.sutd.edu.sg/itrust-labs_datasets/dataset_info/#swat.

[78] Yifan Jia, Jingyi Wang, Christopher M. Poskitt, Sudipta Chattopadhyay, Jun Sun, and Yuqi Chen. 2021. Adversarial attacks and mitigation for anomaly detectors of cyber-physical systems. *International Journal of Critical Infrastructure Protection* 34 (2021), 100452.

[79] Khurum Nazir Junejo. 2020. Predictive safety assessment for storage tanks of water cyber physical systems using machine learning. *Sādhanā* 45, 1 (2020), 1–16.

[80] Khurum Nazir Junejo and Jonathan Goh. 2016. Behaviour-Based Attack Detection and Classification in Cyber Physical Systems Using Machine Learning. In *Proceedings of the 2nd ACM International Workshop on Cyber-Physical System Security* (Xi'an, China) *(CPSS '16)*. Association for Computing Machinery, New York, NY, USA, 34–43. https://doi.org/10.1145/2899015.2899016

[81] Khurum Nazir Junejo and Asim Karim. 2013. Robust personalizable spam filtering via local and global discrimination modeling. *Knowledge and information systems* 34, 2 (2013), 299–334.

[82] Khurum Nazir Junejo and David Yau. 2016. Data Driven Physical Modelling For Intrusion Detection In Cyber Physical Systems. In *Proceedings of the Singapore Cyber-Security Conference (SG-CRC) 2016*. IOS Press, 43 – 57.

[83] Vassilis G Kaburlasos, Eleni Vrochidou, Fotios Panagiotopoulos, Charalampos Aitsidis, and Alexander Jaki. 2019. Time Series Classification in Cyber-Physical System Applications by Intervals' Numbers Techniques. In *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 1–6.

[84] Abdullah Khalili and Ashkan Sami. 2015. SysDetect: A systematic approach to critical state determination for Industrial Intrusion Detection Systems using

Apriori algorithm. *Journal of Process Control* 32 (2015), 154–160.

[85] Levent Koc, Thomas A Mazzuchi, and Shahram Sarkani. 2012. A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier. *Expert Systems with Applications* 39, 18 (2012), 13492–13500.

[86] Nickolaos Koroniotis, Nour Moustafa, and Elena Sitnikova. 2019. Forensics and deep learning mechanisms for botnets in internet of things: A survey of challenges and solutions. *IEEE Access* 7 (2019), 61764–61785.

[87] Dimitrios Kosmanos, Apostolos Pappas, Leandros Maglaras, Sotiris Moschoyiannis, Francisco J Aparicio-Navarro, Antonios Argyriou, and Helge Janicke. 2020. A novel Intrusion Detection System against spoofing attacks in connected Electric Vehicles. *Array* 5 (2020), 100013.

[88] Rafał Kozik, Michał Choraś, Massimo Ficco, and Francesco Palmieri. 2018. A scalable distributed machine learning approach for attack detection in edge computing environments. *J. Parallel and Distrib. Comput.* 119 (2018), 18–26.

[89] Mark A Kramer. 1991. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal* 37, 2 (1991), 233–243.

[90] Philipp Kreimel, Oliver Eigner, and Paul Tavolato. 2017. Anomaly-Based Detection and Classification of Attacks in Cyber-Physical Systems. In *Proceedings of the 12th International Conference on Availability, Reliability and Security* (Reggio Calabria, Italy) *(ARES '17)*. Association for Computing Machinery, New York, NY, USA, Article 40, 6 pages. https://doi.org/10.1145/3098954.3103155

[91] Prashanth Krishnamurthy, Ramesh Karri, and Farshad Khorrami. 2019. Anomaly detection in real-time multi-threaded processes using hardware performance counters. *IEEE Transactions on Information Forensics and Security* 15 (2019), 666–680.

[92] Sudha Krishnamurthy, Soumik Sarkar, and Ashutosh Tewari. 2014. Scalable anomaly detection and isolation in cyber-physical systems using bayesian networks. In *Dynamic Systems and Control Conference*, Vol. 46193. American Society of Mechanical Engineers, V002T26A006.

[93] Ravikiran Krishnan and Sudeep Sarkar. 2015. Conditional distance based matching for one-shot gesture recognition. *Pattern Recognition* 48, 4 (2015), 1298–1310.

[94] Vipin Kumar, Himadri Chauhan, and Dheeraj Panwar. 2013. K-Means Clustering Approach to Analyze NSL-KDD Intrusion Detection Dataset. *International Journal of Soft Computing and Engineering (IJSCE) ISSN* (2013), 2231–2307.

[95] Ajay Kumara and CD Jaidhar. 2018. Automated multi-level malware detection system based on reconstructed semantic view of executables using machine learning techniques at VMM. *Future Generation Computer Systems* 79 (2018), 431–446.

[96] Mehmet Necip Kurt, Oyetunji Ogundijo, Chong Li, and Xiaodong Wang. 2018. Online cyber-attack detection in smart grid: A reinforcement learning approach. *IEEE Transactions on Smart Grid* 10, 5 (2018), 5174–5185.

[97] Y. Kwon, H. K. Kim, Y. H. Lim, and J. I. Lim. 2015. A behavior-based intrusion detection technique for smart grid infrastructure. In *2015 IEEE Eindhoven PowerTech*. IEEE, 1–6.

[98] Jordan Landford, Rich Meier, Richard Barella, Xinghui Zhao, Eduardo Cotilla-Sanchez, Robert B Bass, and Scott Wallace. 2015. Fast Sequence Component Analysis for Attack Detection in Synchrophasor Networks. *arXiv preprint arXiv:1509.05086* (2015).

[99] Adrian P Lauf, Richard A Peters, and William H Robinson. 2010. A distributed intrusion detection system for resource-constrained devices in ad-hoc networks. *Ad Hoc Networks* 8, 3 (2010), 253–266.

[100] James Le. 2017. The 10 Deep Learning Methods AI Practitioners Need to Apply. https://medium.com/cracking-the-data-science-interview/the-10-deep-learning-methods-ai-practitioners-need-to-apply-885259f402c1.

[101] Kun-Lun Li, Hou-Kuan Huang, Sheng-Feng Tian, and Wei Xu. 2003. Improving one-class SVM for anomaly detection. In *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 03EX693)*, Vol. 5. IEEE, 3077–3081.

[102] Peng Li and Oliver Niggemann. 2020. Non-convex hull based anomaly detection in CPPS. *Engineering Applications of Artificial Intelligence* 87 (2020), 103301.

[103] Wenjuan Li, Weizhi Meng, Chunhua Su, and Lam For Kwok. 2018. Towards false alarm reduction using fuzzy if-then rules for medical cyber physical systems. *Ieee Access* 6 (2018), 6530–6539.

[104] Yu Li, Min Li, Bo Luo, Ye Tian, and Qiang Xu. 2020. DeepDyve: Dynamic Verification for Deep Neural Networks. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 101–112.

[105] Yu-Qi Li, Li-Quan Xiao, Jing-Hua Feng, Bin Xu, and Jian Zhang. 2020. AquaSee: Predict Load and Cooling System Faults of Supercomputers Using Chilled Water Data. *Journal of Computer Science and Technology* 35, 1 (2020), 221–230.

[106] Zhai Liang, Tang Xinming, Li Lin, and Jiang Wenliang. 2005. Temporal Association Rule Mining based on T-Apriori Algorithm and its typical application. In *Proceedings of international symposium on spatio-temporal modeling, spatial reasoning, analysis, data mining and data fusion*. Citeseer.

[107] Junyu Lin, Lei Xu, Yingqi Liu, and Xiangyu Zhang. 2020. Composite Backdoor Attack for Deep Neural Network by Mixing Existing Benign Features. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 113–131.

[108] Qin Lin, Sridha Adepu, Sicco Verwer, and Aditya Mathur. 2018. TABOR: A Graphical Model-Based Approach for Anomaly Detection in Industrial Control Systems. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security* (Incheon, Republic of Korea) *(ASIACCS '18)*. Association for Computing Machinery, New York, NY, USA, 525–536. https://doi.org/10.1145/3196494.3196546

[109] Ondrej Linda, Todd Vollmer, and Milos Manic. 2009. Neural network based intrusion detection system for critical infrastructures. In *Neural Networks, International Joint Conference on*. IEEE, 1827–1834.

[110] Robert Lipovsky. 2016. New wave of cyber attacks against Ukrainian power industry. http://www.welivesecurity.com/2016/01/11.

[111] Hongbo Liu, Yan Wang, Jian Liu, Jie Yang, and Yingying Chen. 2014. Practical User Authentication Leveraging Channel State Information (CSI). In *Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security* (Kyoto, Japan) *(ASIA CCS '14)*. Association for Computing Machinery, New York, NY, USA, 389–400. https://doi.org/10.1145/2590296.2590321

[112] George Loukas, Tuan Vuong, Ryan Heartfield, Georgia Sakellari, Yongpil Yoon, and Diane Gan. 2017. Cloud-based cyber-physical intrusion detection for vehicles using deep learning. *IEEE Access* 6 (2017), 3491–3508.

[113] Huimin Lu, Yujie Li, Shenglin Mu, Dong Wang, Hyoungseop Kim, and Seiichi Serikawa. 2017. Motor anomaly detection for unmanned aerial vehicles using reinforcement learning. *IEEE internet of things journal* 5, 4 (2017), 2315–2322.

[114] Yong Luo, Dacheng Tao, Bo Geng, Chao Xu, and Stephen J Maybank. 2013. Manifold regularized multitask learning for semi-supervised multilabel image classification. *Image Processing, IEEE Transactions on* 22, 2 (2013), 523–536.

[115] Yuan Luo, Ya Xiao, Long Cheng, Guojun Peng, and Danfeng Daphne Yao. 2020. Deep Learning-Based Anomaly Detection in Cyber-Physical Systems: Progress and Opportunities. *arXiv preprint arXiv:2003.13213* (2020).

[116] James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1. University of California, Press, Oakland, CA, USA, 281–297.

[117] Leandros A Maglaras and Jianmin Jiang. 2014. Intrusion detection in scada systems using machine learning techniques. In *Science and Information Conference (SAI), 2014*. IEEE, 626–631.

[118] Lorenzo Fernández Maimó, Ángel Luis Perales Gómez, Félix J García Clemente, Manuel Gil Pérez, and Gregorio Martínez Pérez. 2018. A self-adaptive deep learning-based system for anomaly detection in 5G networks. *IEEE Access* 6 (2018), 7700–7712.

[119] Matti Mantere, Mirko Sailio, and Sami Noponen. 2013. Network traffic features for anomaly detection in specific industrial control system network. *Future Internet* 5, 4 (2013), 460–473.

[120] Matti Mantere, Mirko Sailio, and Sami Noponen. 2014. A Module for Anomaly Detection in ICS Networks. In *Proceedings of the 3rd International Conference on High Confidence Networked Systems* (Berlin, Germany) *(HiCoNS '14)*. Association for Computing Machinery, New York, NY, USA, 49–56. https://doi.org/10.1145/2566468.2566478

[121] A. P. Mathur and N. O. Tippenhauer. 2016. 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)*. 31–36. https://doi.org/10.1109/CySWater.2016.7469060

[122] Robert McMillan. 2008. CIA Says Hackers Have Cut Power Grid. https://www.pcworld.com/article/141564/article.html.

[123] Adam McNeil. 2019. All this EternalPetya stuff makes me WannaCry. https://blog.malwarebytes.com/threat-analysis/malware-threat-analysis/2017/07/all-this-eternalpetya-stuff-makes-me-wannacry/.

[124] Trend Micro. 2019. What You Need to Know About the LockerGoga Ransomware. https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/what-you-need-to-know-about-the-lockergoga-ransomware.

[125] Milos Miljanovic. 2012. Comparative analysis of recurrent and finite impulse response neural networks in time series prediction. *Indian J. Comput. Eng* 3, 1 (2012).

[126] Erxue Min, Xifeng Guo, Qiang Liu, Gen Zhang, Jianjing Cui, and Jun Long. 2018. A survey of clustering with deep learning: From the perspective of network architecture. *IEEE Access* 6 (2018), 39501–39514.

[127] JR Minkel. 2008. The 2003 Northeast Blackout–Five Years Later. https://www.scientificamerican.com/article/2003-blackout-five-years-later/.

[128] P. Mishra, P. Aggarwal, A. Vidyarthi, P. Singh, B. Khan, H. Haes Alhelou, and P. Siano. 2021. VMShield: Memory Introspection-based Malware Detection to Secure Cloud-based Services against Stealthy Attacks. *IEEE Transactions on Industrial Informatics* (2021), 1–1. https://doi.org/10.1109/TII.2020.3048791

[129] Robert Mitchell and Ing-Ray Chen. 2014. A Survey of Intrusion Detection Techniques for Cyber-Physical Systems. *ACM Comput. Surv.* 46, 4, Article 55 (March 2014), 29 pages. https://doi.org/10.1145/2542049

[130] Robert Mitchell and Ing-Ray Chen. 2015. Behavior Rule Specification-based Intrusion Detection for Safety Critical Medical Cyber Physical Systems. *Dependable and Secure Computing, IEEE Transactions on* 12, 1 (2015), 16–30.

[131] Thomas Morris, Bradley Srivastava, Anurag adepu2016usingand Reaves, Wei Gao, Kalyan Pavurapu, and Ram Reddi. 2011. A control system testbed to validate critical infrastructure protection concepts. *International Journal of Critical Infrastructure Protection* 4, 2 (2011), 88–103.

[132] Nour Moustafa, Benjamin Turnbull, and Kim-Kwang Raymond Choo. 2018. An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things. *IEEE Internet of Things Journal* 6, 3 (2018), 4815–4830.

[133] Z Muda, W Yassin, MN Sulaiman, and NI Udzir. 2011. Intrusion detection based on K-Means clustering and Naïve Bayes classification. In *Information Technology in Asia (CITA 11), 2011 7th International Conference on*. IEEE, 1–6.

[134] Patric Nader, Paul Honeine, and Pierre Beauseroy. 2013. Intrusion detection in scada systems using one-class classification. In *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European*. IEEE, 1–5.

[135] Patric Nader, Paul Honeine, and Pierre Beauseroy. 2014. -norms in One-Class Classification for Intrusion Detection in SCADA Systems. *Industrial Informatics, IEEE Transactions on* 10, 4 (2014), 2308–2317.

[136] Patric Nader, Paul Honeine, and Pierre Beauseroy. 2014. Mahalanobis-based one-class classification. In *Machine Learning for Signal Processing (MLSP), 2014 IEEE International Workshop on*. 1–6.

[137] Daniel Nahmias, Aviad Cohen, Nir Nissim, and Yuval Elovici. 2019. TrustSign: Trusted Malware Signature Generation in Private Clouds Using Deep Feature Transfer Learning. In *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[138] Anjum Nazir and Rizwan Ahmed Khan. 2021. A novel combinatorial optimization based feature selection method for network intrusion detection. *Computers & Security* 102 (2021), 102164. https://doi.org/10.1016/j.cose.2020.102164

[139] Lily Hay Newman. 2018. Russian Hackers Haven't Stopped Probing the US Power Grid. https://www.wired.com/story/russian-hackers-us-power-grid-attacks/.

[140] Department of Homeland Security. 2020. ICS-CERT Advisories https://ics-cert.us-cert.gov/advisories.

[141] Colin O'Reilly, Alexander Gluhak, and Muhammad Ali Imran. 2016. Distributed anomaly detection using minimum volume elliptical principal component analysis. *IEEE Transactions on Knowledge and Data Engineering* 28, 9 (2016), 2320–2333.

[142] Anton Ivanov Orkhan Mamedov, Fedor Sinitsyn. 2017. Bad Rabbit ransomware. https://securelist.com/bad-rabbit-ransomware/82851/.

[143] Charlie Osborne. 2018. Industroyer: An in-depth look at the culprit behind Ukraine's power grid blackout. https://www.zdnet.com/article/industroyer-an-in-depth-look-at-the-culprit-behind-ukraines-power-grid-blackout/.

[144] Safa Otoum, Burak Kantarci, and Hussein Mouftah. 2019. Empowering reinforcement learning on big sensed data for intrusion detection. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 1–7.

[145] Safa Otoum, Burak Kantarci, and Hussein T Mouftah. 2019. On the feasibility of deep learning in sensor network intrusion detection. *IEEE Networking Letters* 1, 2 (2019), 68–71.

[146] Koyena Pal, Sridhar Adepu, and Jonathan Goh. 2017. Effectiveness of association rules mining for invariants generation in cyber-physical systems. In *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*. IEEE, 124–127.

[147] José M Balbuena Palácios, Jorge R Beingolea Garay, Alexadre M Oliveira, and Sergio T Kofuji. 2013. Intrusion Detection System: A hybrid approach for Cyber-Physical Environments. *Technology* 39 (2013), 193–204.

[148] Shengyi Pan, Thomas Morris, and Uttam Adhikari. 2015. Developing a hybrid intrusion detection system using data mining for power systems. *Smart Grid, IEEE Transactions on* 6, 6 (2015), 3104–3113.

[149] Mrutyunjaya Panda and Manas Ranjan Patra. 2009. Ensembling rule based classifiers for detecting network intrusions. In *Advances in Recent Technologies in Communication and Computing, 2009. ARTCom'09. International Conference on*. IEEE, 19–22.

[150] Martina Panfili, Alessandro Giuseppi, Andrea Fiaschetti, Homoud B Al-Jibreen, Antonio Pietrabissa, and Franchisco Delli Priscoli. 2018. A game-theoretical approach to cyber-security of critical infrastructures based on multi-agent reinforcement learning. In *2018 26th Mediterranean Conference on Control and Automation (MED)*. IEEE, 460–465.

[151] Sudeep Pasricha, Janardhan Rao Doppa, Krishnendu Chakrabarty, Saideep Tiku, Daniel Dauwe, Shi Jin, and Partha Pratim Pande. 2017. Special session paper: data analytics enables energy-efficiency and robustness: from mobile to manycores, datacenters, and networks. In *2017 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ ISSS)*. IEEE, 1–10.

[152] Ahmed Patel, Hitham Alhussian, Jens Myrup Pedersen, Bouchaib Bounabat, Joaquim Celestino Júnior, and Sokratis Katsikas. 2017. A nifty collaborative intrusion detection and prevention architecture for smart grid ecosystems. *Computers & Security* 64 (2017), 92–109.

[153] Xi Peng, Jiashi Feng, Shijie Xiao, Jiwen Lu, Zhang Yi, and Shuicheng Yan. 2017. Deep sparse subspace clustering. *arXiv preprint arXiv:1709.08374* (2017).

[154] Nicole Perlroth. 2012. In Cyberattack on Saudi Firm, U.S. Sees Iran Firing Back. https://www.nytimes.com/2012/10/24/business/global/cyberattack-on-saudi-oil-firm-disquiets-us.html.

[155] Abigail Pichel. 2014. HAVEX Targets Industrial Control Systems. https://www.trendmicro.com/vinfo/us/threat-encyclopedia/web-attack/139/havex-targets-industrial-control-systems.

[156] David Martin Powers. 2011. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies* 2, 1 (2011), 37–63.

[157] J. R. Quinlan. 1996. Improved Use of Continuous Attributes in C4.5. *J. Artif. Int. Res.* 4, 1 (March 1996), 77–90.

[158] Ragunathan Rajkumar. 2012. A cyber–physical future. *Proc. IEEE* 100, Special Centennial Issue (2012), 1309–1312.

[159] MR Gauthama Raman, Nivethitha Somu, Sahruday Jagarapu, Tina Manghnani, Thirumaran Selvam, Kannan Krithivasan, and VS Shankar Sriram. 2019. An efficient intrusion detection technique based on support vector machine and improved binary gravitational search algorithm. *Artificial Intelligence Review* 53 (2019), 1–32.

[160] Heena Rathore, Chenglong Fu, Amr Mohamed, Abdulla Al-Ali, Xiaojiang Du, Mohsen Guizani, and Zhengtao Yu. 2018. Multi-layer security scheme for implantable medical devices. *Neural Computing and Applications* 32 (2018), 1–14.

[161] Thomas Roccia. 2018. Triton Malware Spearheads Latest Attacks on Industrial Systems. https://www.mcafee.com/blogs/other-blogs/mcafee-labs/triton-malware-spearheads-latest-generation-of-attacks-on-industrial-systems/.

[162] Lior Rokach and Oded Maimon. 2005. Clustering methods. In *Data mining and knowledge discovery handbook*. Springer, 321–352.

[163] Bernardino Romera-Paredes and Philip H. S. Torr. 2015. An Embarrassingly Simple Approach to Zero-Shot Learning. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37 (ICML'15)*. JMLR.org, Lille, France, 2152–2161.

[164] Shailendra Sahu and BM Mehtre. 2015. Network intrusion detection system using J48 Decision Tree. In *Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on*. IEEE, 2023–2026.

[165] Naoum Sayegh, Imad H Elhajj, Ayman Kayssi, and Ali Chehab. 2014. SCADA Intrusion Detection System based on temporal behavior of frequent patterns. In *Mediterranean Electrotechnical Conference (MELECON), 2014 17th IEEE*. IEEE, 432–438.

[166] Homeland Security. 2020. ICS Cybersecurity Landscape for Managers (FRE2115 R00). https://ics-cert-training.inl.gov/learn/course/external/view/elearning/59/ICSCybersecurityLandscapeforManagersFRE2115R00.

[167] Sohil Atul Shah and Vladlen Koltun. 2018. Deep continuous clustering. *arXiv preprint arXiv:1803.01449* (2018).

[168] Raheel Shaikh. 2018. Feature Selection Techniques in Machine Learning with Python. https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e.

[169] Sparsh Sharma and Ajay Kaul. 2018. Hybrid fuzzy multi-criteria decision making based multi cluster head dolphin swarm optimized IDS for VANET. *Vehicular Communications* 12 (2018), 23–38.

[170] Mansour Sheikhan and Zahra Jadidi. 2014. Flow-based anomaly detection in high-speed links using modified GSA-optimized neural network. *Neural Computing and Applications* 24, 3-4 (2014), 599–611.

[171] Alex Shenfield, David Day, and Aladdin Ayesh. 2018. Intelligent intrusion detection systems using artificial neural networks. *ICT Express* 4, 2 (2018), 95–99.

[172] Jianhua Shi, Jiafu Wan, Hehua Yan, and Hui Suo. 2011. A survey of cyber-physical systems. In *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*. IEEE, 1–6.

[173] Sooyeon Shin, Taekyoung Kwon, Gil-Yong Jo, Youngman Park, and Haekyu Rhy. 2010. An experimental study of hierarchical intrusion detection for wireless industrial sensor networks. *Industrial Informatics, IEEE Transactions on* 6, 4 (2010), 744–757.

[174] Jill Slay and Michael Miller. 2008. *Lessons learned from the maroochy water breach*. Springer.

[175] Richard Socher, Milind Ganjoo, Christopher D. Manning, and Andrew Y. Ng. 2013. Zero-Shot Learning through Cross-Modal Transfer. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1* (Lake Tahoe, Nevada) *(NIPS'13)*. Curran Associates Inc., Red Hook, NY, USA, 935–943.

[176] Alexander N. Sokolov, Andrey N. Ragozin, Ilya A. Pyatnitsky, and Sergei K. Alabugin. 2019. Applying of Digital Signal Processing Techniques to Improve the Performance of Machine Learning-Based Cyber Attack Detection in Industrial Control System. In *Proceedings of the 12th International Conference on Security of Information and Networks* (Sochi, Russia) *(SIN '19)*. Association for Computing Machinery, New York, NY, USA, Article 23, 4 pages. https://doi.org/10.1145/3357613.3357637

[177] Saleh Soltan, Prateek Mittal, and H Vincent Poor. 2019. Line failure detection after a cyber-physical attack on the grid using bayesian regression. *IEEE*

Transactions on Power Systems 34, 5 (2019), 3758–3768.

[178] Robin Sommer and Vern Paxson. 2010. Outside the closed world: On using machine learning for network intrusion detection. In *Security and Privacy (SP), 2010 IEEE Symposium on*. IEEE, 305–316.

[179] Jungsuk Song, Kenji Ohira, Hiroki Takakura, Yasuo Okabe, and Yongjin Kwon. 2008. A clustering method for improving performance of anomaly-based intrusion detection system. *IEICE transactions on information and systems* 91, 5 (2008), 1282–1291.

[180] Jungsuk Song, Hiroki Takakura, Yasuo Okabe, and Yongjin Kwon. 2009. Unsupervised anomaly detection based on clustering and multiple one-class SVM. *IEICE transactions on communications* 92, 6 (2009), 1981–1990.

[181] Jost Tobias Springenberg. 2015. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390* (2015).

[182] Melissa Stockman, Dipankar Dwivedi, Reinhard Gentz, and Sean Peisert. 2019. Detecting control system misbehavior by fingerprinting programmable logic controller functionality. *International Journal of Critical Infrastructure Protection* 26 (2019), 100306.

[183] Gayathri Sugumar and Aditya Mathur. 2019. A method for testing distributed anomaly detectors. *International Journal of Critical Infrastructure Protection* 27 (2019), 100324. https://doi.org/10.1016/j.ijcip.2019.100324

[184] Yixin Sun, Kangkook Jee, Suphannee Sivakorn, Zhichun Li, Cristian Lumezanu, Lauri Korts-Parn, Zhenyu Wu, Junghwan Rhee, Chung Hwan Kim, Mung Chiang, et al. 2020. Detecting Malware Injection with Program-DNS Behavior. In *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 552–568.

[185] RS Sutton and AG Barto. 1998. Introduction to Reinforcement Learning. MIT Press. *Cambridge, MA* (1998).

[186] Christopher T. Symons and Justin M. Beaver. 2012. Nonparametric Semi-Supervised Learning for Network Intrusion Detection: Combining Performance Improvements with Realistic in-Situ Training. In *Proceedings of the 5th ACM Workshop on Security and Artificial Intelligence* (Raleigh, North Carolina, USA) *(AISec '12)*. Association for Computing Machinery, New York, NY, USA, 49–58. https://doi.org/10.1145/2381896.2381905

[187] Kerry Tomlinson. 2017. Computer guy who sabotaged his own factory heads to prison. https://archerint.com/computer-guy-sabotaged-factory-heads-prison/.

[188] Chih-Fong Tsai, Yu-Feng Hsu, Chia-Ying Lin, and Wei-Yang Lin. 2009. Intrusion detection by machine learning: A review. *Expert Systems with Applications* 36, 10 (2009), 11994–12000.

[189] Muhammad Azmi Umer, Chuadhry Mujeeb Ahmed, Muhammad Taha Jilani, and Aditya P. Mathur. 2021. Attack Rules: An Adversarial Approach to Generate Attacks for Industrial Control Systems using Machine Learning. arXiv:cs.CR/2107.05127

[190] Muhammad Azmi Umer, Aditya Mathur, Khurum Nazir Junejo, and Sridhar Adepu. 2017. Integrating Design and Data Centric Approaches to Generate Invariants for Distributed Attack Detection. In *Proceedings of the 2017 Workshop on Cyber-Physical Systems Security and PrivaCy* (Dallas, Texas, USA) *(CPS '17)*. Association for Computing Machinery, New York, NY, USA, 131–136. https://doi.org/10.1145/3140241.3140248

[191] Muhammad Azmi Umer, Aditya Mathur, Khurum Nazir Junejo, and Sridhar Adepu. 2020. Generating Invariants using Design and Data-centric Approaches for Distributed Attack Detection. *International Journal of Critical Infrastructure Protection* 28 (2020), 100341.

[192] Muhammad Azmi Umer, Aditya Mathur, Khurum Nazir Junejo, and Sridhar Adepu. 2020. A method of generating invariants for distributed attack detection, and apparatus thereof. US Patent App. 16/754,732.

[193] Sharmila Wagh, Anagha Khati, Auzita Irani, Naba Inamdar, and Rashmi Soni. 2014. Effective Framework of J48 Algorithm using Semi-Supervised Approach for Intrusion Detection. *International Journal of Computer Applications* 94, 12 (2014).

[194] Jingyu Wang, Dongyuan Shi, Yinhong Li, Jinfu Chen, Hongfa Ding, and Xianzhong Duan. 2018. Distributed framework for detecting PMU data manipulation attacks with deep autoencoders. *IEEE Transactions on Smart Grid* 10, 4 (2018), 4401–4410.

[195] Xueyang Wang, Charalambos Konstantinou, Michail Maniatakos, Ramesh Karri, Serena Lee, Patricia Robison, Paul Stergiou, and Steve Kim. 2016. Malicious firmware detection with hardware performance counters. *IEEE Transactions on Multi-Scale Computing Systems* 2, 3 (2016), 160–173.

[196] Yi Wang, Mahmoud M Amin, Jian Fu, and Heba B Moussa. 2017. A novel data analytical approach for false data injection cyber-physical attack mitigation in smart grids. *IEEE Access* 5 (2017), 26022–26033.

[197] Xiaotao Wei, Houkuan Huang, and Shengfeng Tian. 2007. A grid-based clustering algorithm for network anomaly detection. In *The First International Symposium on Data, Privacy, and E-Commerce (ISDPE 2007)*. IEEE, 104–106.

[198] Sean Whalen, Nathaniel Boggs, and Salvatore J. Stolfo. 2014. Model Aggregation for Distributed Content Anomaly Detection. In *Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop* (Scottsdale, Arizona, USA) *(AISec '14)*. Association for Computing Machinery, New York, NY, USA, 61–71. https:

//doi.org/10.1145/2666652.2666660

[199] Dumidu Wijayasekara, Ondrej Linda, Milos Manic, and Craig Rieger. 2014. FN-DFE: fuzzy-neural data fusion engine for enhanced resilient state-awareness of hybrid energy systems. *Cybernetics, IEEE Transactions on* 44, 11 (2014), 2065–2075.

[200] Di Wu, Fan Zhu, and Ling Shao. 2012. One shot learning gesture recognition from rgbd images. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*. IEEE, 7–12.

[201] Liyuan Xiao, Yetian Chen, and Carl K Chang. 2014. Bayesian Model Averaging of Bayesian Network Classifiers for Intrusion Detection. In *Computer Software and Applications Conference Workshops (COMPSACW), 2014 IEEE 38th International*. IEEE, 128–133.

[202] Jun Yan, Haibo He, Xiangnan Zhong, and Yufei Tang. 2016. Q-learning-based vulnerability analysis of smart grid against sequential topology attacks. *IEEE Transactions on Information Forensics and Security* 12, 1 (2016), 200–210.

[203] Weizhong Yan, Lalit K Mestha, and Masoud Abbaszadeh. 2019. Attack Detection for Securing Cyber Physical Systems. *IEEE Internet of Things Journal* 6, 5 (2019), 8471–8481.

[204] Bo Yang, Xiao Fu, Nicholas D. Sidiropoulos, and Mingyi Hong. 2017. Towards K-Means-Friendly Spaces: Simultaneous Deep Learning and Clustering. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70* (Sydney, NSW, Australia) *(ICML'17)*. JMLR.org, 3861–3870.

[205] Dayu Yang, Alexander Usynin, and J Wesley Hines. 2006. Anomaly-based intrusion detection for SCADA systems. In *5th intl. topical meeting on nuclear plant instrumentation, control and human machine interface technologies (npic&hmit 05)*. Citeseer, 12–16.

[206] Nong Ye, Syed Masum Emran, Qiang Chen, and Sean Vilbert. 2002. Multivariate statistical analysis of audit trails for host-based intrusion detection. *Computers, IEEE Transactions on* 51, 7 (2002), 810–820.

[207] Honggang Yu, Kaichen Yang, Teng Zhang, Yun-Yun Tsai, Tsung-Yi Ho, and Yier Jin. 2020. Cloudleak: Large-scale deep learning models stealing through adversarial examples. In *Proceedings of Network and Distributed Systems Security Symposium (NDSS)*.

[208] Liu Yuxun and Xie Niuniu. 2010. Improved ID3 algorithm. In *2010 3rd International Conference on Computer Science and Information Technology*, Vol. 8. IEEE, 465–468.

[209] Bo Zhang, Chunxia Dou, Dong Yue, and Zhanqiang Zhang. 2018. Response hierarchical control strategy of communication data disturbance in micro-grid under the concept of cyber physical system. *IET Generation, Transmission & Distribution* 12, 21 (2018), 5867–5878.

[210] Bin Zhang, Jia-Hai Yang, Jian-Ping Wu, and Ying-Wu Zhu. 2012. Diagnosing traffic anomalies using a two-phase model. *Journal of Computer Science and Technology* 27, 2 (2012), 313–327.

[211] Jiaheng Zhang, Zhiyong Fang, Yupeng Zhang, and Dawn Song. 2020. Zero Knowledge Proofs for Decision Tree Predictions and Accuracy. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 2039–2053.

[212] Yingwei Zhang, Yuanjian Fu, Zhenbang Wang, and Lin Feng. 2017. Fault detection based on modified kernel semi-supervised locally linear embedding. *IEEE access* 6 (2017), 479–487.

[213] Yichi Zhang, Lingfeng Wang, Weiqing Sun, Robert C Green, Mansoor Alam, et al. 2011. Artificial immune system based intrusion detection in a distributed hierarchical network architecture of smart grid. In *Power and Energy Society General Meeting, 2011 IEEE*. IEEE, 1–8.

[214] Yichi Zhang, Lingfeng Wang, Weiqing Sun, Robert C Green, Mansoor Alam, et al. 2011. Distributed intrusion detection system in a multi-layer network architecture of smart grids. *Smart Grid, IEEE Transactions on* 2, 4 (2011), 796–808.

[215] Yang Zhong, Hirohumi Yamaki, and Hiroki Takakura. 2011. A grid-based clustering for low-overhead anomaly intrusion detection. In *2011 5th International Conference on Network and System Security*. IEEE, 17–24.

[216] B. Zhu and S. Sastry. 2010. SCADA-specific Intrusion Detection / Prevention Systems : A Survey and Taxonomy.

[217] Giulio Zizzo, Chris Hankin, Sergio Maffeis, and Kevin Jones. 2019. Adversarial machine learning beyond the image domain. In *2019 56th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 1–4.