international journal of critical infrastructure protection 38 (2022) 100516

Contents lists available at ScienceDirect

International Journal of Critical Infrastructure Protection

journal homepage: www.elsevier.com/locate/ijcip

Machine learning for intrusion detection in industrial control systems: Applications, challenges, and recommendations

Muhammad Azmi Umer [a,b,*], Khurum Nazir Junejo [c], Muhammad Taha Jilani [b], Aditya P. Mathur [d]

[a] DHA Suffa University, DG-78 Off Khayaban-e-Tufail, Phase 7 (Ext.) Defence Housing Authority, Karachi, Pakistan
[b] Karachi Institute of Economics and Technology, PAF Base Korangi Creek, Karachi, Pakistan
[c] DNNae Inc., Islamabad, Pakistan
[d] iTrusts Centre for Research in Cyber Security, Singapore University of Technology and Design, 8 Somapah Rd, Singapore

ARTICLE INFO

Keywords:
Machine learning
Deep learning
Intrusion detection
Anomaly detection
Cyber-attacks
Cyber physical systems
Critical infrastructures
IoT
Industrial Control Systems

ABSTRACT

Methods from machine learning are used in the design of secure Industrial Control Systems. Such methods focus on two major areas: 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 for intrusion and anomaly detection, namely, supervised, semi-supervised, unsupervised, and reinforcement learning. The literature available in the public domain was carefully selected, analyzed, and placed along a 10-dimensional space for ease of comparison. This multi-dimensional approach is found valuable in the comparison of the methods considered and enables a scientific discussion on their utility in specific environments. The challenges associated in using machine learning, and gaps in research, are identified and recommendations made.

1. Introduction

Progress in machine learning (ML), coupled with attempted and successful cyber-attacks on critical infrastructure, has sparked a wave of interest in behavior-based Intrusion Detection Systems (IDS) for Industrial Control Systems (ICS). This article is a survey of methods from ML that are applied to detect intrusions and anomalies in ICS. Our focus is on a comparison of the proposed approaches and their usability in operational environments. With this focus 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 Cyber–Physical Systems (CPS). Specifically, works surveyed in this article are placed along a 10-dimensional space as described in Section 5. Use of the proposed 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. The systems of interest in this survey are primarily those where an ICS controls a physical process. Such systems are constituents of critical infrastructure in a city 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 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 [1–3]. While ICS remains the focus of this survey, we have not avoided references to systems that do not use ICS but fall in the CPS category.

1.1. 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 [4,5]. 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.

* Corresponding author at: DHA Suffa University, DG-78 Off Khayaban-e-Tufail, Phase 7 (Ext.) Defence Housing Authority, Karachi, Pakistan.
E-mail addresses: muhammadazmiumer@yahoo.com (M.A. Umer), junejo@gmail.com (K.N. Junejo), m.taha@kiet.edu.pk (M.T. Jilani), aditya_mathur@sutd.edu.sg (A.P. Mathur).

https://doi.org/10.1016/j.ijcip.2022.100516
Received 30 July 2021; Received in revised form 11 January 2022; Accepted 5 February 2022
Available online 17 February 2022
1874-5482/

1.2. 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 [6] 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 [7]. A water treatment plant in the U.S. was attacked in 2006 [8]. 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.

1.3. 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.

1.4. 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 be rapidly 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 new literature in this area will continue to be added with suitable comments. Tables in this article that place each research publication in a 10-dimensional space will be kept at this site and updated.

1.5. 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 literature surveyed here focuses mostly on plants controlled by an ICS. Literature related to detection of anomalies in network traffic is generally classified under the "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 term "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 several such abbreviations. The abbreviations used in this article are listed in Table 8.

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

1.6. Organization

The process used in collecting articles referenced in this survey is summarized in Section 2. Intrusion Detection Systems are introduced and categorized broadly in Section 3. There exist other surveys that also report on the ML techniques applied to ICS. Such surveys are cited, and differences from this survey identified, in Section 4. Literature surveyed and evaluated is placed along a 10-dimensional space described in Section 5. Methods from ML 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, 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 contains a summary of this survey work and offers conclusions.

2. Collection of articles

Articles considered 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 2019/2020 were retrieved in multiple iterations as described in Fig. 1. In the first iteration, a breadth-first study of each article was performed to extract various properties including the domain, approach, limitations, strengths, etc. We did not select articles that do not define any specific ML approach in IDS, and instead offer only a general discussion on ML approaches. 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 articles published in journals and magazines. 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 in the second.

3. Intrusion detection systems

Intrusion detection systems (IDS) 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. Before diving into a detailed survey, we first discuss the basic structure of IDS and its various types.

An IDS consists of five key components. These components involve Data collection devices, Knowledge Base, Detector, Configuration device, and a Response component [25]. Data collection is done using sensors. The knowledge base contains the rules or information depending on the type of IDS. The IDS is a key component that varies significantly. The detector is responsible for identifying the intrusion using the Knowledge base and the data obtained from sensors. Configuration devices provide the current state of IDS, and the Response component is responsible to initiate appropriate actions based on detected intrusions. In the current study, three types of IDS are considered, namely, signature-based, specification-based, and behavior-based.

Table 1
Incidents on industrial control systems.

Year	Incident	Year	Incident
2019	LockerGoga ransomware [9]	2014	Port Hudson paper mill insider threat [10]
2018	Olympic destroyer [11]	2013	Havex [12]
2018	TRITON triconex SIS malfunction [13]	2012	Shamoon [14]
2017	TEMP.isotope campaign [15]	2011	Duqu [16]
2017	BadRabbit ransomware [17]	2010	Stuxnet [6]
2017	EternalPetya ransomware [18]	2008	CIA reports foreign utilities hacked [19]
2017	WannaCry ransomware [20]	2007	Aurora generator test [21]
2016	Industroyer ukraine blackout [22]	2003	Northeast blackout [23]
2015	BlackEnergy 3 ukraine blackout [24]	2001	Maroochy sewage spill [7]

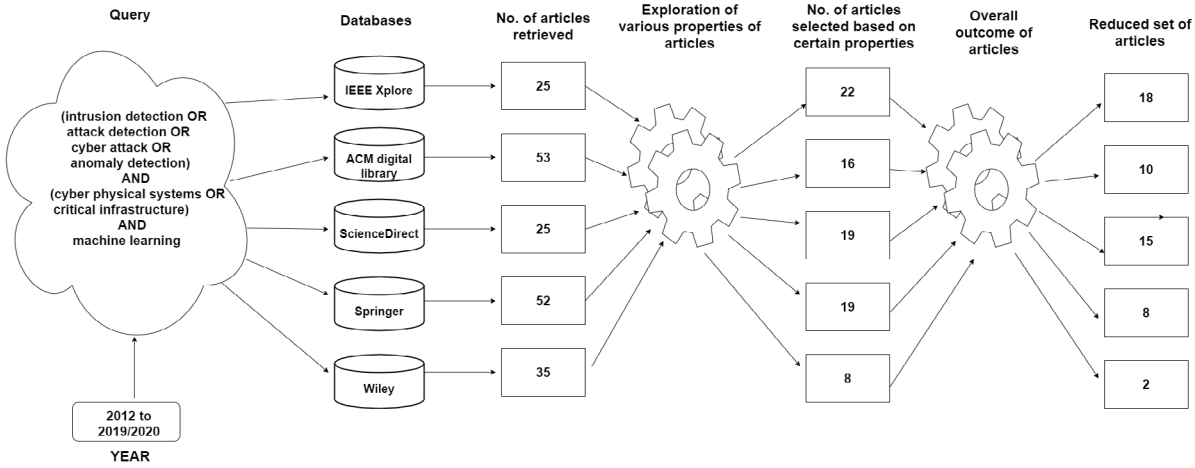

Fig. 1. Retrieval and selection of articles.

3.1. Signature-based IDS

This type of IDS requires a predefined dictionary of attack patterns. These attack patterns are placed in the knowledge base of the IDS. This knowledge base is used to detect an intrusion if any pattern detected during plant operation matches one or more of the predefined attack patterns [26]. 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 signatures in complex physical processes. There are numerous ways to automate the generation of malware signatures. For example, in the study reported in [27] 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.

3.2. Specification-based IDS

This approach develops a mathematical model to define the normal operation of the physical process under consideration. This mathematical model is placed in the knowledge base of IDS. An anomaly is said to exist whenever the process deviates from the prediction by the predefined model [28]. 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 [29] 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 [30] also uses 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.

3.3. 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. The trained model is then placed in the knowledge base of IDS. It is later used by the detector to detect intrusions. This approach is favored against incorrect vendor specifications as it trains the model on empirical data [31] and thus enables the identification of incorrect vendor specifications. For instance, the study reported in [32] noticed different levels of a water tank in a water treatment plant. According to the vendor specifications, the upper bound 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 rely on statistical techniques [33] 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

Table 2
Comparison with past surveys.

Past surveys	Difference
[34]	Main theme is to shift current ICS to cloud based infrastructure.
[35]	Focus on IDS in general terms; not specifically on ICS.
[36]	Focus on Deep Learning (DL) techniques with types of anomalies, evaluation metrics, strategies, and implementation details; different taxonomy
[37]	A general survey of physics-based attack detection in CPS; not focused on ML.
[38]	A survey of IDS in CPS focusing only on detection technique and Audit material.
[39,40]	A survey of CPS discussing challenges and future trends; does not focus on IDS approaches for CPS.
[41]	A survey of IDS based only on Supervised machine learning approaches
[42–46]	Focus on Network-based IDS.
[47]	Focus on Reinforcement Learning (RL) based Q-learning methods for securing CPS.
[48]	Focus on SCADA specific intrusion detection and prevention.

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
Algorithms	Algorithms used for training the ML model
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
Data type	Type of dataset used; Actual or Simulated
Data availability	Dataset used for model training is available or not
Metrics	Metrics used for evaluating the effectiveness of the ML techniques used

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 in the deployment of IDS based on these techniques in complex 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.

4. Related surveys

A comparison of related surveys is presented in Table 2. Three main attributes are used in this survey, i.e., Intrusion detection, Machine Learning (ML), and Industrial Control Systems (ICS). There exist surveys that focus on intrusion detection but are not related to ML or ICS [37,49]. Similarly, there exist surveys that focus on intrusion detection using machine learning but do not focus on ICS [50,51]. There exist surveys where machine learning is discussed as a component of cybersecurity, for example, a survey of ICS security that discussed ML is reported in [34]. This article discusses the benefits and shortcomings of using ML techniques for detecting anomalies in ICS. However, the primary theme of this survey is the need for shifting current ICS to a cloud-based infrastructure. This survey has minimally discussed machine learning-based IDS; and offers only an overview of machine learning approaches. Similarly, there is another survey [41] focusing on Machine Learning for SCADA security. This survey has only focused on Supervised Learning and uses fewer dimensions than used here.

Deep Learning-based IDS are discussed in [35]. This work focuses on intrusion detection in its general terms and does not focus 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 [36]. A taxonomy was developed for the survey that includes type of anomalies, evaluation metrics, strategies, and implementation details.

A survey of physics-based anomaly detection is reported in [37]. 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 [38]. 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 which are discussed in Section 5.

A survey of CPS is reported in [39,40]. This survey discusses the challenges and future research trends but does not focus on IDS. Network-based IDS is surveyed in [42–46], though the authors do not address the scenario that differs from conventional networks. A survey reported in [47] 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 [48] focuses on SCADA specific intrusion detection and prevention. This survey focuses on behavior-based approaches for intrusion detection in CPS focusing on ML and DL techniques. Such approaches have recently gained 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

Given the vast amount of literature available in the domain of ML-based IDS, it is of value to compare them. With this as our goal, a multi-dimensional approach was adopted to categorize the relevant literature. We target the work on IDS for ICS but have not avoided some references to a broader class of CPS. Specifically, works surveyed in this article are placed along a 10-dimensional space where the dimensions are domain, audit material, complexity, algorithms, feature selection, time series, dataset, data type, data availability, 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 know that the adoption of a multi-dimensional approach for categorization of research has also been adopted by other researchers [38]. However, the multi-dimensional space adopted in this survey is richer in the dimensions selected. These are enumerated in Table 3 and described in the following subsections.

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
[52]	Conveyor belt system	Physical	Simple	k-NN, and NB	Yes	Yes	Annon	Actual	No	Confidence, Accuracy
[53]	Chemical plant	Physical	Simple	OCSVM	Yes	Yes	HITL	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[54]	Annon	Physical	Simple	OCSVM	Yes	Annon	Annon	Actual	No	FPR, FNR
[55]	Gas, and Water	Physical	Simple	SVDD, and KPCA	No	Yes	MSU, and UCI	Actual	Yes	Accuracy
[56]	Water	Physical	Simple	LSTM	Yes	Yes	SWaT	Actual	Yes	Accuracy
[57]	Industrial demonstrator	Physical	Simple	OCSVM, DINA	No	Yes	Industrial demonstrator, and Wind Turbines	Actual, and Simulated	No	TPR, TNR, F1 score, and Balanced Accuracy
[58]	Water, Gas, and Energy	Network	Simple	ESNN, SOCCADF, OCC-SVM, OCC-CD/CPE	No	Yes	Water, Gas, and Electric	Actual	Yes	TPR, TNR, TA, Precision, Recall, and F1-score
[59]	Annon	Network	Hybrid	SVM	No	Yes	Annon	Actual	No	DR, and IR
[60]	Energy	Physical	Hybrid	DAE, OCSVM, AdaBoost + C4.5, XGBoost, MLP, SVM, k-NN.	Yes	Yes	Annon	Simulated	No	Accuracy, Precision, Recall, and F1 score
[61]	Energy	Physical	Simple	GDLM, SVM, MLP, and PCA	No	Yes	Annon	Simulated	No	F1 score

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 labeled as Annon in Tables 4, 5, 6, and 7 do not specify the domain in which the proposed approach is applied, instead they mention it as "some CPS/ICS".

In our survey we discovered that the least explored ICS in smart utility is gas. Even though a few such ICS are listed in Table 7, they rely on a relatively simple gas ICS testbed at Mississippi State University (MSU) [124], which consists of 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 [125], BACnet [115], and DNP3 [126]. Hence, it is commercially viable to develop IDS for such protocols. A study reported in [114] used bits per packet, connections per second, and recent/mean interval time and count of Goose messages for this purpose. Another study reported in [98] used several responses against a command to detect attacks. The study in [111,121] uses deep packet inspection to compute the n-gram features from the payload of the packet. The method used in this work is to construct 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 [115] uses Ethernet, IP, UDP, and BACnet packet header attributes to train its IDS. The study reported in [104] 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 a physical process controlled by an ICS is challenging as the components, their size, and functionality of each sub-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 a model: voltage phase angle, voltage magnitude, current phase angle, current phase magnitude, zero voltage phase angle magnitude, current phase angle magnitude, the relay frequency, frequency delta for relays, apparent impedance and angle observed by relays, status flags for relays, snort alert status for each relay, control panel remote trip status, and their correlations [94,95]. A study reported in [32] 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, the authors in [116] use 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 [97,98] uses a couple of physical process features together with several network traffic features to detect attacks in gas ICS. Another study reported in [122] uses CPU and OS usage parameters in addition to network traffic features to detect attacks in a simulated CPS made up of different SUN Microsystem servers and workstations. A study reported in [103] used Wireshark to capture network logs and physical stream data such as temperature and airflow. This data is 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 feature space where a classifier is trained for improved performance [115], or (b) multiple classifiers are trained separately but cooperate to arrive at a decision [93,96,115]. A study reported in [91] first used the K-means to cluster the data

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
[62]	Energy	Physical	Simple	MSA, SVM, and ANN	No	Yes	PMU	Actual, and Simulated	No	Accuracy
[63]	Healthcare	Physical	Simple	k-NN, NN, SVM, DT, NB, and ZeroR.	No	Yes	Annon	Actual, and Simulated	No	Accuracy, Precision, Recall, and F1 score
[64]	Water	Network, and Physical	Hybrid	SVM, and SMC	Yes	Yes	Annon	Actual	No	Accuracy, Sensitivity, and Specificity
[65]	Smart home	Network	Simple	NB, BN, J48, Zero R, OneR, Logistic, SVM, MLP, and RF	Yes	Yes	Annon	Actual	No	F1 score
[66]	Energy	Physical	Simple	BR with ARD	No	Yes	Annon	Simulated	No	FP, FN, and PT
[67]	Gas, and Energy	Physical	Simple	ELM	Yes	Yes	Annon	Actual, and Simulated	No	ROC, TPR, and FPR
[68]	Water	Network, and Physical	Simple	SVM	Yes	Yes	SWaT, and WADI	Actual	Yes	Accuracy
[69]	Annon	Physical	Simple	CNN	Yes	Yes	Annon	Actual	No	Accuracy
[70]	Water	Network, and Physical	Simple	RF, NBTree, LMT, J48, PART, MLP, HTree, LogF, and SVM.	Yes	Yes	SWaT	Actual	Yes	Precision, and Sensitivity
[71]	Water	Physical	Simple	NN	Yes	Yes	SWaT	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[72]	Electric vehicles	Network, and Physical	Hybrid	RF, and k-NN	No	Yes	Annon	Simulated	No	Accuracy, DR, ROC, and AUC
[73]	Annon	Physical	Hybrid	LSTM, NN, SVC, and SVM	Yes	Yes	Annon	Simulated	No	Probability of detection
[74]	Annon	Network	Simple	ANN	No	Yes	Annon	Actual	No	Accuracy, Precision, Sensitivity, and ROC
[75]	Cloud	Physical	Simple	LR, RF, NB, RT, SMO, and J48	Yes	Yes	Annon	Actual	No	TPR, TNR, F1 score, and Accuracy
[76]	Energy	Network	Simple	SVM	No	Yes	Annon	Actual	No	Accuracy, and AUC
[77]	Drones	Physical	Simple	GA, XGBoost, and SVM	No	Yes	Annon	Actual	No	Precision
[78]	Energy	Physical	Simple	SVM, k-NN, RF, and CNN	Yes	Yes	Annon	Actual	No	Accuracy
[79]	Cloud	Network	Simple	ELM	Yes	Yes	CTU	Actual	Yes	TPR, FPR, TNR, FNR, Precision, Accuracy, ER, F1 score, MC
[80]	VANETs	Network	Simple	SVM	Yes	Yes	Annon	Simulated	No	DE, FPR, DT and CH Load
[81]	Annon	Network	Simple	AE, LSTM, MLP, SVM, LDA and QDA	Yes	Yes	NSL-KDD	Actual	Yes	Precision, Recall, F1 score, and Accuracy
[82]	Annon	Network	Simple	BN, NB, MLPNN, J48, and SVM	Yes	Yes	NSL-KDD CUP, and UNSW-NB15	Actual	Yes	DR, FAR, and Accuracy

followed by self-organizing maps (SOM) for final classification. Another study reported in [93] 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 [96]. 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. The third tier uses an artificial neural network to forecast the value of each sensor based on the historical data. Lastly, fuzzy logic is used to raise an alarm based on the outputs of tier 2 and tier 3. A study reported an anomaly-based IDS for SCADA [91] 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 specific threshold.

5.4. Algorithms

Machine learning and deep learning algorithms can be broadly classified into three major categories, namely, Supervised Learning, Unsupervised Learning, and Reinforcement Learning. We have not included the Semi-Supervised learning as Supervised learning algorithms can be used in Semi-Supervised learning. The details of each category is discussed in Sections 7, 8, and 10.

5.5. 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 [127]. Deep Learning techniques do not require explicit feature selection because they have an inherent capability to select the best features. A study reported in [128] proposed a feature selection

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
[83]	Annon	Network	Simple	MLP	Yes	Yes	KDD Cup 99, NSL-KDD, SCX2012, and UNSW-NB15	Actual	Yes	DR, FAR, and AR
[84]	Energy	Physical	Simple	RF, OneR, JRip, Adaboost + JRip, SVM, and NN	Yes	Yes	MSU, and ORNL	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[85]	Supercomputer/ Water	Physical	Simple	LSTM	No	Yes	Tianhe-1A	Actual	Yes	RMSE, and Accuracy
[86]	Healthcare	Physical	Simple	MLP, and SVM	Yes	No	ECG-ID	Actual	Yes	Accuracy, Precision, Recall, and F1 score
[87]	Annon	Network	Simple	MLP, MGSA, PSO, and EBP	No	Yes	Intrusion Detection dataset	Actual	Yes	CCR, ER, MR, and FAR
[88]	Annon	Network	Simple	ASCH-IDS, and RBC-IDS	Yes	No	KDD'99 Dataset	Simulated	Yes	AR, FNR, DR, ROC , and F1 score
[89]	Energy	Network, and Physical	Hybrid	BPNN, and ELM	Yes	Yes	Annon	Simulated	No	Error/Hz
[90]	Vehicle	Network, and Physical	Simple	RNN, MLP, LR, DT(5.0), RF, and SVM	Yes	Yes	Annon	Actual	No	Accuracy
[91]	Annon	Network	Hybrid	k-means-SOM	No	No	KDDCup1999	Actual	Yes	FPR, TPR, and DR
[92]	Energy	Network	Simple	SVM, and AIS	No	No	KDDCup1999	Simulated	Yes	FPR, FNR, and No. of Detections
[93]	Energy	Physical	Hybrid	Ensemble of SVMs	No	Yes	Bonneville Power Administration	Actual	No	Recall, Precision, F1 score, and Latency
[94]	Energy	Physical	Hybrid	CPM	No	No	MSU Power	Simulated	Yes	Accuracy, and FPR
[95]	Energy	Physical	Simple	NB, OneR, Nnge, Jripper, RF, SVM, and Adaboost	No	No	MSU Power	Simulated	Yes	F1 score
[96]	Energy	Physical	Hybrid	Fuzzy-neural data fusion engine	No	No	Idaho National Labs energy sys. model	Actual	No	Error Graphs
[97]	Gas	Hybrid	Simple	NB, OneR, Nnge, RF, SVM, and J48	No	No	MSU	Actual	Yes	Precision, and Recall
[98]	Water	Hybrid	Simple	NN	No	No	MSU	Actual	No	Accuracy, FP, and FN
[32]	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
[99]	Water	Physical	Simple	RTI+, and BN	Yes	Yes	SWaT	Actual	Yes	CP, and PS

method based on Tabu Search and Random Forest. They used Tabu Search for searching and Random Forest as a learning algorithm for intrusion detection.

5.6. 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 appropriate in the ML or DL model. A study reported in [129] 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. The effectiveness of the proposed approach was tested using a benchmark classification problem.

5.7. 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 their classifiers [93,109], thus making the realism and fidelity questionable. Other works resort to validate models on simulated data [94,95,112,122]. Some studies have used the NSL-KDD99 dataset [130] 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) .[2] Their power system dataset is a simulated smart grid data consisting of data under normal behavior, attacks, and faults. This dataset was used in [94,95] for intrusion detection in ICS using ML. Their water storage tank and gas pipeline dataset is developed using a small scale laboratory testbed and used in [55,97,116] to detect intrusions in CPS using ML. 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 [131] is another publicly available dataset from 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 carried out by running the plant non-stop for eleven consecutive days. For the first seven days the plant was run in a normal state while

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

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
[100]	Annon	Network	Simple	k-means, and SVM	No	Yes	Annon	Actual	No	ADR, and Accuracy
[101]	Water	Physical	Simple	Apriori	No	Yes	SWaT	Actual	Yes	Accuracy
[55,102]	Gas, and Water	Physical	Simple	OCSVM	No	No	MSU	Actual	Yes	Accuracy
[103]	HVAC	Hybrid	Simple	BN	No	No	Annon	Actual	No	Accuracy
[104,105]	Printed intelligence	Network	Hybrid	SOM	No	No	PrintoCent	Actual	No	None
[106]	Energy	Physical	Simple	IF, PCA, SVM, k-NN, NB, and MLP	No	Annon	SE-MF	Actual	No	Accuracy, and F1 score
[107]	Water	Network	Simple	k-means, and LOF	Yes	Yes	Annon	Actual	No	PLC scan time
[108]	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
[110]	Water	Physical	Simple	Apriori	No	No	Annon	Actual	No	Accuracy
[111]	Annon	Network	Simple	PAYL, POSEIDON, Anagram, McPAD	No	No	Annon	Actual	No	FPR, and DR
[112]	Annon	Network	Simple	Multi hop clustering	No	No	Annon	Simulated	No	DR
[113]	Aviation, and Robots	Network	Hybrid	Statistical	No	No	Annon	Simulated	No	% of Devient Nodes for convergence
[114]	Energy	Network	Simple	Statistical	No	No	Korean substation	Actual	No	Precision, Recall, F1 score, FPR, and FNR
[115]	Energy	Network	Hybrid	Bayesian	No	Yes	American University of Beirut power plant	Actual	No	Accuracy, and FP
[116]	Gas	Physical	Simple	OCSVM	No	No	MSU	Actual	Yes	Accuracy
[117–119]	Water	Physical	Simple	FP-growth	Yes	Yes	SWaT	Actual	Yes	Accuracy
[120]	Energy	Physical	Simple	k-means	No	Yes	PeCanStreet Project, and Irish Social Science Data Archive	Actual	Yes	Accuracy
[121]	SCADA	Network	Simple	Statistical	No	No	AUT09	Actual	No	DR, and FP
[122]	SCADA	Hybrid	Simple	Statistical	Yes	Yes	Annon	Simulated	No	Detection diagrams
[123]	SCADA	Network	Simple	OCSVM	No	No	Annon	Actual	No	Accuracy

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

Term*	Expansion	Term*	Expansion
AE	Autoencoder	**LDA**	Linear Discriminant 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		

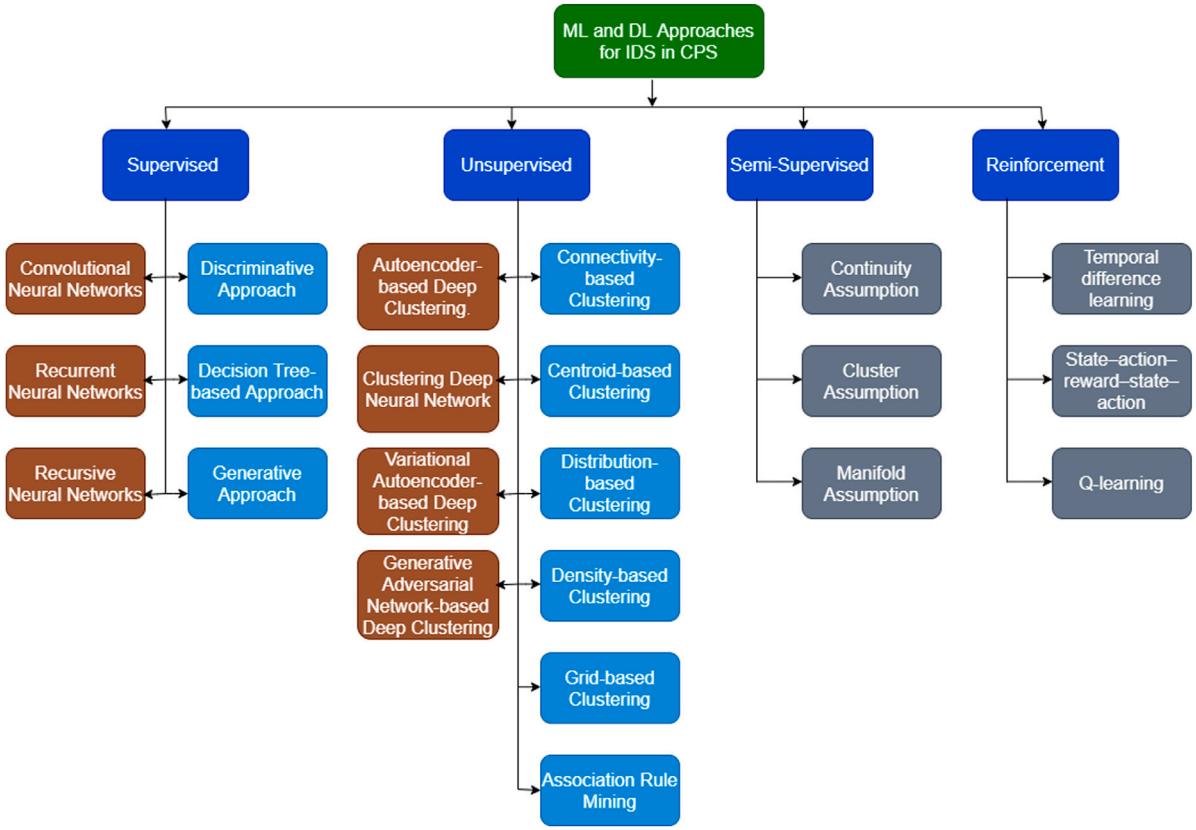

Fig. 2. Categorization of machine learning approaches for detecting intrusions in Industrial Control Systems.

during the last four days specifically crafted attacks were launched on the plant. Therefore, this dataset contains both the normal and attack data from an operational testbed. Both network and physical process data were collected for this purpose. Following the publication of the dataset in [131], iTrust has made public several other datasets collected from the SWaT testbed [132]. The SWaT datasets have been used in a large number of research projects including, though not limited to, [31,117,118,133–135].

5.8. Data type

The datasets used in the literature are of two types: Actual and Simulated. Here 'Actual' refers to the operational data of a physical operational plant, or a testbed, while 'Simulated' refers to the synthetic data created by researchers for experimentation. Most of the studies reported in this survey used the actual data while some reported the use of simulated data. Most of the simulated data is used in the domain of energy [60,66]. A few studies have used both the actual and simulated data, as for example in [57,62].

5.9. Data availability

Data availability is necessary to replicate an experiment reported publicly. Therefore, while categorizing the research work we have carefully analyzed the availability of the dataset used including both free and paid datasets.

5.10. 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 report an accuracy higher than 90%. Therefore, accuracy is not adequate to assess the performance of an 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 adequate 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 precision.

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 fewer false positives (FP) whereas, recall is the number of attacks detected by the classifier. A high value of recall leads to fewer false negatives (FN). An ideal classifier is associated with high precision and recall. The F-Measure enables combining both into a single metric, i.e., the harmonic mean of precision and recall. It is a more conservative measure than the arithmetic mean. 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 is 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 robust for highly skewed problems [136]. The reason for their robustness is that by

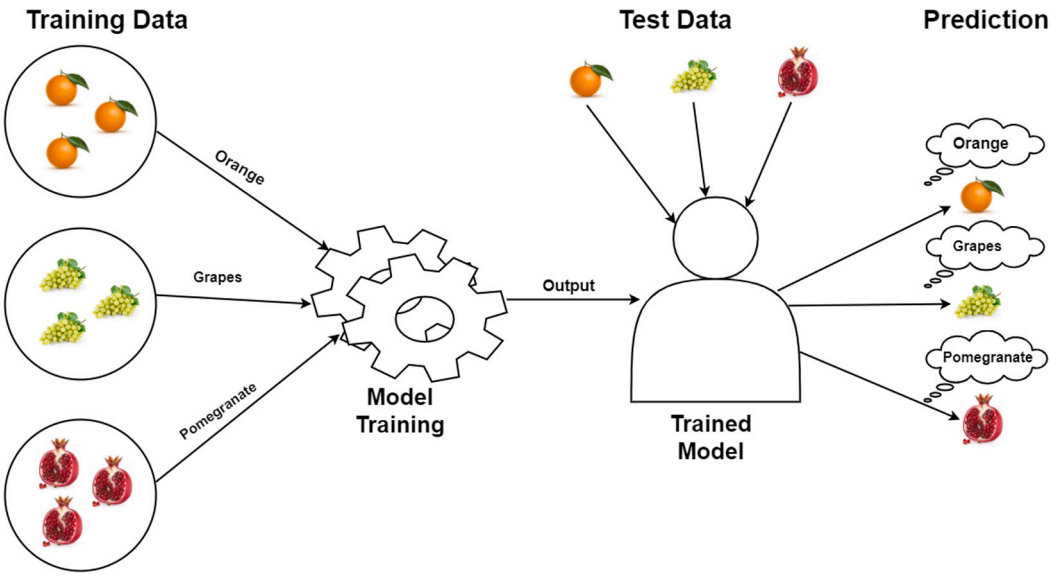

Fig. 3. Supervised Learning.

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 [32,72,76,137,138] have used this measure for evaluating their respective classifiers.

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 [32,93,118]. Among these, the authors in [93] reported the latency. They define latency as the number of cycles after data spoofing begins but before the classifier correctly identifies a string of 30 consecutive cycles as spoofed. The authors in [32,118] reported the time to detect an attack while those in [32] also reported the percentage of time the attack was detected during its existence.

6. Machine learning approaches for intrusion detection

As shown in Fig. 2, ML and DL techniques can be classified into four major categories, namely, 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. The difference between the first three approaches lies in whether 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 whereas unlabeled training data is plenty and available. All areas mentioned in Fig. 2 are discussed in subsequent sections.

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 [55,102]. 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 [32] showed that the 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 much explored for IDS in ICS are one-shot learning [139,140] and zero-shot learning [141,142]. One-shot learning refers to a scenario where only one instance of each attack type is available in the labeled training data. 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 a set of zero-day attacks. Thus, the performance of this type of learning depends 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 the unsupervised approaches. This approach can also 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 shown in Fig. 3. For each instance of the training dataset, SL uses 'n' features from feature vector 'X', i.e., $[x_1, x_2 \ldots, x_n]$ to learn the class variable, or label '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. Once the model is trained it is used to predict the labels on test 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 Tables 5 and 6

7.1. Supervised learning approaches

Certain behavior-based approaches have used conventional statistical techniques [114,122]. These approaches use metrics such as the

mean and standard deviation on sensor measurements. These techniques are not fully automatable 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; details of each are given below.

7.1.1. Discriminative approaches

Support Vector Machines (SVM) are linear classifiers, non-probabilistic, and perform binary classification. When using SVM, the data points are projected to a higher dimensional feature space. Then, a hyperplane is learned to distinguish the data points from the two classes. The goal of learning a hyperplane is to enlarge the difference between the closest data points of the classes and thereby provide stronger generalization on the unobserved data. This property of SVM makes it robust for classification problems including IDS [143].

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 [144].

Instance-based learning algorithms (IBK) do not work on generalization 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 algorithm. IBK has been used in [91,145,146] for IDS. The Non-Nested Generalized Exemplars (NNGE) also belong to this class of algorithms [147]. It was applied to detect network intrusions in KDDCup 1999 dataset [130].

Artificial immune systems try to mimic the complex vertebrate immune system [148]. They are intelligent and robust computing systems. Fuzzy rules were developed in [96] to express the normal behavior of the system using Fuzzy-Neural Data Fusion Engine (FN-DFE). Subsequently these fuzzy rules were used for anomaly detection by comparing them with previously described rules that govern the system behavior. A classifier based on neural networks was used to make the concluding decision based on the observed anomalies.

Multinomial Logistic Regression (LR) is comparable to linear regression and serves as an alternative to Linear Discriminant Analysis (LDA). However, they both have different underlying assumptions. LR assumes Bernoulli distribution while linear regression assumes Gaussian distribution. LR uses the logistic function 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 found as successful in IDS [149].

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. The decisions follow a tree-like structure having 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 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 [150,151]. 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 [152]. 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 accuracy in each 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 [153,154]. An ensemble method (AdaBoost) was used in [155] for intrusion detection in the network traffic of IoT devices. While IoT devices are playing a vital role in providing comfort in 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 [156,157]. 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 acceptable generalization over the unobserved data. The model is learned in a single iteration over the training data.

BayesNet are directed acyclic graphs [158] that represent the set of random variables along with their conditional dependencies. The dependency between the variables can be eliminated by connecting them through an edge. In the real-world, the attributes of a dataset are likely dependent on each other thus making BayesNet approach superior to NB in 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 a large feature space [159].

Due to their simplicity, performance, and low computational complexity, Bayesian classifiers are commonly used to solve real-world problems. A study reported in [99] 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 [75] 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 that focuses on the Artificial Neural Network (ANN). It does not require a complex set of features to be manually engineered by humans, instead it aims to learn these features. This makes ANN a promising approach for ICS due to its unique dynamics. Deep learning is capable of dealing with high-velocity data [160], thus making it desirable for ICS. However, computational complexities associated with deep learning [161] add to challenges in its use in practice.

7.2.1. Convolutional neural networks

Convolutional Neural Network (CNN) is a type of deep neural network. It is often used on visual images. CNN is a variant of Multi-Layer Perceptron (MLP). One of the important properties of MLP is its Fully Connectedness in which every single neuron in one layer has connectivity with all neurons in the next layer. This may lead to overfitting. Regularization methods are used to reduce the extent of overfitting. Regularization methods include adjustments of weights to configure the loss function. These methods exploit the underlying hierarchical pattern in the data to derive complex patterns from relatively small and simple patterns. CNN has been used in a classification model for different PLC programs using data generated from the phasor measurement unit (PMU) during the execution of PLC programs [78]. CNN was also used for anomaly detection in the PMU data. CNN was used to detect keystrokes using sensor data from a nearby mobile phone [162]. In this work keystrokes were classified using a real-world dataset from 20-users. CNN was used for anomaly detection using thermal side channels in [69]. Thermal images were captured on a predefined time window and used in CNN to detect anomalies using information from a predefined actual active time.

7.2.2. Recurrent and recursive neural networks

Recurrent Neural Networks (RNN) are a class of ANNs. Its edges input the next time step instead of the next layer of the current time step [163]. RNNs refer to two classes of networks, namely, finite impulse and infinite impulse networks. Both classes have temporal dynamic behavior [164] and could have additional stored states where storage would be controlled by the neural network. RNNs are also a type of deep neural network. They apply the same set of weights recursively over a structured input sequence. Therefore, they lead to structured predictions over variable-sized input sequences. Compared to ANN, RNNs work hierarchically on the input sequence [163].

A study reported in [90] makes use of RNN to protect vehicles from cyber-attacks. All computations are performed in the cloud. Long short-term memory (LSTM) networks are a type of RNN. They are found effective in speech recognition [165]. 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 [56] LSTM is used to train a model on the normal data from a water treatment plant and its performance tested on attack data. In this study the adaptive attacks were launched 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 [81]. A layered architecture using different ML techniques for proactive fault management by predicting sensor values at different stages is proposed in [73]. A hybrid machine learning approach is 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 is proposed for this purpose and serves as the quorum for the final decision of the model.

Accurate prediction of faults in supercomputers can be used to overcome financial losses. Historical chilled water dataset is used in [85] to predict the load of a supercomputer using LSTM. Later, a Z-score model of predicted values is used to identify the anomalies. 5G Networks pose a formidable challenge to the security of data. Though 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 [166] 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 is to classify the anomalous data at the high velocity to cope with higher velocity data of 5G Networks and therefore accuracy is not the major concern in this phase. In the subsequent phase, the output of DBN is used by LSTM to recognize the temporal patterns of cyber-attacks.

8. Unsupervised learning

Unsupervised Learning (UL) uses features from feature vector 'X', without the corresponding class variable or label as indicated in Fig. 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 metrics. Literature focusing on the use of UL for IDS 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 wherein a hierarchy of clusters is formed. The method can be divided into the following two categories [167]: 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 eventually forming 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 a 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 from 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 [120] for the classification of compromised meters in an Advanced Metering Infrastructure (AMI). AMIs are 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 load balancing and other critical functions in a power grid. A consensus correction scheme is introduced in [120] 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 [58] 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 in PLCs, a mechanism to augment PLCs with AES-256 Encryption and Decryption is proposed in [107]. 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 [100] uses the Channel State Information (CSI) to identify a 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. In this technique objects belonging to the same distribution are assigned to the same cluster. The technique often suffers from overfitting unless constraints are applied to model complexity. Gaussian mixture model is applied in [61] 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), an unsupervised machine learning technique, was used

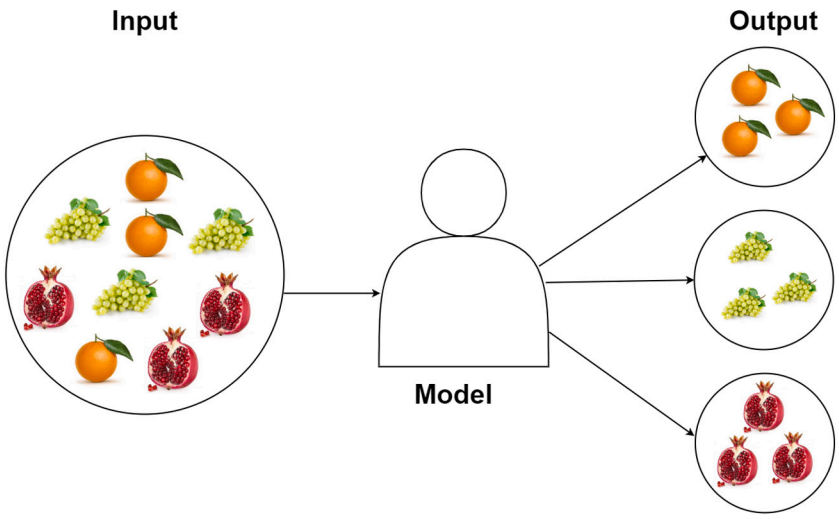

Fig. 4. Unsupervised Learning.

for dimensionality reduction. The performance of the 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 in the dataset could be detected whether or not it is an anomaly. The proposed method was also compared with Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP). Overall, the study reported has an acceptable 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 from the remainder. 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 [168]. It works on the principle of "Density-reachability" using a distance threshold. DBSCAN is used in [169] 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, is proposed in [170] 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 multi-dimensional datasets [171]. 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 is used in [172] for anomaly detection. The authors evaluated the system using Kyoto2006+ and the KDD Cup 1999 datasets. False Positive rate of the proposed algorithm was better than the Song based K-means [173], Song based One-Class SVM [174], Y-means [175], k-means [176], and Li [177]. To partition high dimensional and large data space, a grid-based algorithm is proposed in [178]. The algorithm works in two phases. In the first phase, 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 [179] is a rule-based machine learning technique 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 an itemset which is a set of values of one or more attributes. Itemsets that meet the support threshold are referred to 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 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 rules such as $X \implies Y$, where X is an antecedent and Y is the consequent. Rules that satisfy the confidence threshold are qualified for the final set of association rules.

ARM is used in [110] to determine the critical system state for the intrusion detection system using the Apriori algorithm. At the same time, it also incorporates 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 [101] to generate invariants for a water treatment plant using the Apriori algorithm. This was 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, as reported in [117–119] 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 [132]. The invariants were also placed as 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 originating between the application layer to the kernel layer get recorded

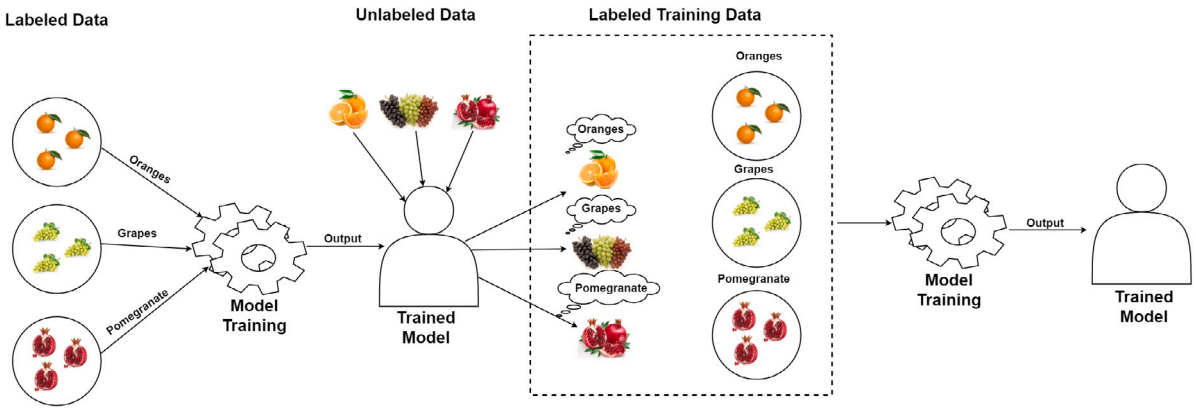

Fig. 5. Semi-Supervised Learning.

in system logs and traces. These logs and traces are helpful in monitoring the performance of any system and are for anomaly detection. However, these traces and logs are generally large in a real-time system, and therefore online anomaly detection remains a challenge. A deep recursive attentive model (DReAM) is proposed in [180] 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 at 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 is proposed in [161] 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 [181]. It first learns the representation, i.e., encoding from data and is then used for dimensionality reduction. It thus trains the network to ignore noise and attempts to learn a representation close enough to the original input while minimizing the reconstruction loss. There are several AE-based deep clustering methods, for example, Deep Clustering Network (DCN) and [182], Deep Embedding Network (DEN) [183]. Both these methods use the autoencoder with the k-means clustering algorithm. The former is capable of simultaneously performing feature learning and clustering while the latter is good for clustering-friendly representation [184]. Similarly, Deep Embedded Regularized Clustering (DEPICT) [185], and Deep Continuous Clustering (DCC) [186] also use the autoencoders to perform clustering. The former works well on the image data while the latter does not require any prior knowledge related to cluster number [184].

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 [184].

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 multiple Bayesian methods [184]. It can use stochastic gradient descent [187] and standard backpropagation [188] 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 [184]. 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) [189], Categorical Generative Adversarial Network (CatGAN) [190], and Information Maximizing Generative Adversarial Network (InfoGAN) [191]. DAC uses the Adversarial Autoencoder to perform the clustering while CatGAN uses the Generative Adversarial Network to perform clustering by choosing the prior number of categories rather than the two categories. Similarly, InfoGAN is also used for clustering and as well as for learning the disentangled representations [184].

9. Semi-Supervised Learning

Semi-Supervised Learning (SSL) uses both the labeled and unlabeled data for training of the model. One way using SSL is described in Fig. 5. In the first phase, the model is trained using labeled data as in supervised learning. In the second phase, labels are assigned to the unlabeled data using the trained model 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.

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 [192–194]. A study reported in [195] proposed a model to extract the behavioral patterns of malware using semi-supervised and unsupervised machine learning techniques. SSL is used in [196] to automatically update the attack detection system of CPS using the unlabeled malware data. In the first stage, it captures malware patterns from unlabeled data using UL. Next, 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 [197] 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 [198–200] have shown that semi-supervised approaches can perform better than supervised and unsupervised approaches for conventional network intrusion detection though 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 the Reward. As illustrated in Fig. 6, an agent performs an action in the environment for which it receives some 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 trained using bootstrapping using the current estimate of the value function. Methods are sampled from the environment and updates are performed based on the current estimate [201].

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 in this 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 is used in [202] 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 is proposed in [203] for anomaly detection in the smart grid. The authors proposed an RL based solution to the Partially Observable Markov Decision Process (POMDP) problem. For the optimal defense of CPS in [204], the problem is formulated as a two-player zero-sum game. Deep RL is used to tune the actor–critic Neural Network structure. Likewise in [205], a multi-agent general sum game is used to model the attack problem. RL is 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 [206]. Q-Learning based vulnerability assessment of smart grid is reported in [207] 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 results from the IEEE 5-bus, RTS-79, and IEEE 300-bus systems-based simulation. RL is also used in [208] 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 while Adversarial machine learning is being used to counter its benefits. For example, in [56] LSTM is 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. They have become vulnerable to different adversarial attacks. In this case, the adversary attempts to steal the model by querying the Application Programming Interface (API). For example a study reported in [209] uses an attack methodology to extract the DNN models from various cloud-based platforms. For this purpose, the authors used various algorithms including active and transfer learning. Similarly, in [210] composite attacks are launched using Trojan triggers to disrupt the performance of the DNN model. The Trojan triggers were composed of benign features of multiple labels. The model misclassifies the output when input is stamped with Trojan trigger.

A few studies have tried to tackle the above mentioned challenge though additional work is needed. For example, the authors in [211] used zero knowledge proofs for decision tree. Accuracy and predictions on public dataset are reported without leaking any information about the model. Using the proposed study a decision tree with a depth of 23 and 1029 nodes can generate the zero knowledge proofs in 250 s. Likewise, the authors in [212] use 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. This makes it difficult to detect zero-day attacks. For example, the model proposed in [27] 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 [213] uses the unsupervised machine learning approach to generate attacks for a real-world ICS. The authors used association rule mining to generate attack patterns. Normally, Supervised learning approaches lack attack data, hence the study reported in [213] 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 [214] for SWaT [215]. 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 an effective strategy to defend against zero-day attacks because of the comprehensive attack dictionary created. Similarly, a study reported in [216] uses 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 [215] and WADI [217].

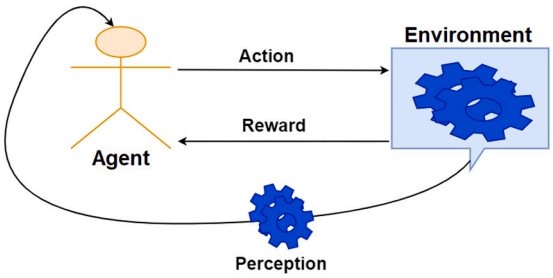

Fig. 6. Reinforcement Learning.

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 favored against incorrect vendor specifications due to its dependability on empirical data but there are issues in 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 [132,215]. Therefore it is a challenge to detect attacks in the heterogeneous physical processes controlled by an ICS. For example, in SWaT testbed [132] an attack on one stage can disrupt the sub-processes in other stages as well. Hence developing a model which can capture the behavior of heterogeneous physical processes remains a major challenge. Though IDS based on physical processes have received relatively little attention, there is a growing trend to detect intrusion in such systems.

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 report a higher accuracy. Therefore, accuracy is not enough to assess the performance of an IDS. It is also important to note that acceptable values of the metrics discussed in Section 5.10 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, 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

An attacker with deep insights would likely render the system vulnerable to stealthy attacks. Such attacks gradually disrupt the performance of the operational plant. Though there are a number of studies addressing this issue, but the detection of such attacks is remains a challenge. For example, a study proposed in [218] uses the Profile-DNS for detecting the stealthy attacks by characterizing the expected DNS behavior. Likewise, the authors in [219] use VMshield for securing cloud platforms against stealthy attacks. They use feature selection using meta-binary particle swarm optimization (BPSO) algorithms. Random Forest was used 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 [101, 110,117,118]. Despite this, there remain gaps that need to be filled. For example, all the studies reported in [101,110,117,118] use data from an ICS controlling a water plant. Though, the authors in [118] 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, [118] uses a time series data and the FP-Growth algorithm to mine the rules. FP-Growth is time agnostic and therefore appears promising to use for Temporal Association Rule Mining [220].

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 dynamics of an ICS. Some studies have reported the use of DL to secure ICS as discussed in Sections 7.2 and 8.2 most of which are SL approaches. There exist only a few studies [161,180] 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 an 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 that works on the basis of agent and environment interaction is the least explored area for securing ICS though a few studies have been reported 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 those mentioned above, there are other promising ML approaches that need to be explored for intrusion detection in ICS. Zero-shot learning [141,142] is one such promising approach for detecting zero-day attacks. Domain adaptation can help learn an IDS for an ICS using data from another ICS. Lastly, distribution shift techniques can be explored for making the model adapt to the changing behavior of the system. 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 unable to find a comparison with the specification or signature-based techniques except in [117,118] where the behavior-based approach is compared with the specification based approach. A comparison of the three types of approaches is needed 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 5.10. 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 common 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, an 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 [103] has reported root cause isolation. Lastly, few studies have modeled the problem using time series, whereas, many ICS repeat the same operations over and over again. More research is 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. 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. Moreover, due to a drastic increase in adversarial machine learning approaches, there is a crucial requirement for defense mechanisms against the machine learning models working in IDS. 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. This area is in a need of a high fidelity benchmark dataset. In brief, 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. 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.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge the time and efforts of Mr. Bilal Hayat Butt for providing valuable suggestions and feedback on the survey.

References

[1] José Barbosa, Paulo Leitão, Damien Trentesaux, Armando W. Colombo, Stamatis Karnouskos, Cross benefits from cyber-physical systems and intelligent products for future smart industries, in: 2016 IEEE 14th International Conference on Industrial Informatics (INDIN), IEEE, 2016, pp. 504–509.
[2] Ragunathan Rajkumar, A cyber–physical future, Proc. IEEE 100 (Special Centennial Issue) (2012) 1309–1312.
[3] Radhakisan Baheti, Helen Gill, Cyber-physical systems, Impact Control Technol. 12 (1) (2011) 161–166.
[4] Homeland Security, ICS Cybersecurity Landscape for Managers (FRE2115 R00), ICS-CERT Virtual Learning Portal (VLP), 2020, https://ics-cert-training.inl.gov/learn/course/external/view/elearning/59/ICSCybersecurityLandscapeforManagersFRE2115R00.
[5] Department of Homeland Security, ICS-CERT Advisories, US Department of Homeland Security, 2020, https://ics-cert.us-cert.gov/advisories.
[6] Nicolas Falliere, Liam O. Murchu, Eric Chien, W32. Stuxnet Dossier, Vol. 5, White Paper, Symantec Corp., Security Response, 2011.
[7] Jill Slay, Michael Miller, Lessons Learned from the Maroochy Water Breach, Springer, 2008.
[8] Alvaro A. Cárdenas, Saurabh Amin, Zong-Syun Lin, Yu-Lun Huang, Chi-Yen Huang, Shankar Sastry, Attacks against process control systems: Risk assessment, detection, and response, in: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, in: ASIACCS '11, Association for Computing Machinery, New York, NY, USA, 2011, pp. 355–366.
[9] Trend Micro, What you need to know about the LockerGoga ransomware, 2019, https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/what-you-need-to-know-about-the-lockergoga-ransomware.
[10] Kerry Tomlinson, Computer guy who sabotaged his own factory heads to prison, 2017, https://archerint.com/computer-guy-sabotaged-factory-heads-prison/.
[11] GReAT, Hades, the actor behind olympic destroyer is still alive, 2018, https://securelist.com/olympic-destroyer-is-still-alive/86169/.
[12] Abigail Pichel, HAVEX targets industrial control systems, 2014, https://www.trendmicro.com/vinfo/us/threat-encyclopedia/web-attack/139/havex-targets-industrial-control-systems.
[13] Thomas Roccia, Triton malware spearheads latest attacks on industrial systems, 2018, https://www.mcafee.com/blogs/other-blogs/mcafee-labs/triton-malware-spearheads-latest-generation-of-attacks-on-industrial-systems/.
[14] Nicole Perlroth, In cyberattack on Saudi firm, U.S. Sees iran firing back, 2012, https://www.nytimes.com/2012/10/24/business/global/cyberattack-on-saudi-oil-firm-disquiets-us.html.
[15] Lily Hay Newman, Russian hackers haven't stopped probing the US power grid, 2018, https://www.wired.com/story/russian-hackers-us-power-grid-attacks/.
[16] GReAT, The Mystery of Duqu 2.0: a sophisticated cyberespionage actor returns, 2015, https://securelist.com/the-mystery-of-duqu-2-0-a-sophisticated-cyberespionage-actor-returns/70504/.
[17] Anton Ivanov Orkhan Mamedov, Bad Rabbit ransomware, 2017, https://securelist.com/bad-rabbit-ransomware/82851/.
[18] Adam McNeil, All this EternalPetya stuff makes me WannaCry, 2019, https://blog.malwarebytes.com/threat-analysis/malware-threat-analysis/2017/07/all-this-eternalpetya-stuff-makes-me-wannacry/.
[19] Robert McMillan, CIA says hackers have cut power grid, 2008, https://www.pcworld.com/article/141564/article.html.
[20] Josh Fruhlinger, What is WannaCry ransomware, how does it infect, and who was responsible?, 2018, https://www.csoonline.com/article/3227906/what-is-wannacry-ransomware-how-does-it-infect-and-who-was-responsible.html.
[21] INCIBE, Aurora vulnerability: origin, explanation and solutions, 2019, https://www.incibe-cert.es/en/blog/aurora-vulnerability-origin-explanation-and-solutions.
[22] Charlie Osborne, Industroyer: An in-depth look at the culprit behind Ukraine's power grid blackout, 2018, https://www.zdnet.com/article/industroyer-an-in-depth-look-at-the-culprit-behind-ukraines-power-grid-blackout/.
[23] J.R. Minkel, The 2003 Northeast blackout–five years later, 2008, https://www.scientificamerican.com/article/2003-blackout-five-years-later/.
[24] Robert Lipovsky, New wave of cyber attacks against Ukrainian power industry, 2016, http://www.welivesecurity.com/2016/01/11.

[25] Aleksandar Lazarevic, Vipin Kumar, Jaideep Srivastava, Intrusion detection: A survey, in: Managing Cyber Threats, Springer, 2005, pp. 19–78.

[26] Wei Gao, Thomas H. Morris, On cyber attacks and signature based intrusion detection for MODBUS based industrial control systems, J. Digit. Forensics Secur. Law 9 (1) (2014) 37–56.

[27] Daniel Nahmias, Aviad Cohen, Nir Nissim, Yuval Elovici, TrustSign: Trusted malware signature generation in private clouds using deep feature transfer learning, in: 2019 International Joint Conference on Neural Networks (IJCNN), IEEE, 2019, pp. 1–8.

[28] Robert Mitchell, Ing-Ray Chen, Behavior rule specification-based intrusion detection for safety critical medical cyber physical systems, Dependable Secure Comput. IEEE Trans. 12 (1) (2015) 16–30.

[29] Sridhar Adepu, Aditya Mathur, Using process invariants to detect cyber attacks on a water treatment system, in: Jaap-Henk Hoepman, Stefan Katzenbeisser (Eds.), ICT Systems Security and Privacy Protection, Springer International Publishing, Cham, 2016, pp. 91–104.

[30] R. Berthier, W.H. Sanders, Specification-based intrusion detection for advanced metering infrastructures, in: 2011 IEEE 17th Pacific Rim International Symposium on Dependable Computing, 2011, pp. 184–193.

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

[32] Khurum Nazir Junejo, David Yau, Data driven physical modelling for intrusion detection in cyber physical systems, in: Proceedings of the Singapore Cyber-Security Conference (SG-CRC) 2016, IOS Press, 2016, pp. 43–57.

[33] Nong Ye, Syed Masum Emran, Qiang Chen, Sean Vilbert, Multivariate statistical analysis of audit trails for host-based intrusion detection, Comput. IEEE Trans. 51 (7) (2002) 810–820.

[34] Deval Bhamare, Maede Zolanvari, Aiman Erbad, Raj Jain, Khaled Khan, Nader Meskin, Cybersecurity for industrial control systems: A survey, Comput. Secur. 89 (2020) 101677.

[35] A.M. Aleesa, B.B. Zaidan, A.A. Zaidan, Nan M. Sahar, 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.

[36] Yuan Luo, Ya Xiao, Long Cheng, Guojun Peng, Danfeng Daphne Yao, Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities, 2020, arXiv preprint arXiv:2003.13213.

[37] Jairo Giraldo, David Urbina, Alvaro Cardenas, Junia Valente, Mustafa Faisal, Justin Ruths, Nils Ole Tippenhauer, Henrik Sandberg, Richard Candell, A survey of physics-based attack detection in cyber-physical systems, ACM Comput. Surv. 51 (4) (2018).

[38] Robert Mitchell, Ing-Ray Chen, A survey of intrusion detection techniques for cyber-physical systems, ACM Comput. Surv. 46 (4) (2014).

[39] Rachana Ashok Gupta, Mo-Yuen Chow, Networked control system: overview and research trends, Ind. Electron. IEEE Trans. 57 (7) (2010) 2527–2535.

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

[41] Jakapan Suaboot, Adil Fahad, Zahir Tari, John Grundy, Abdun Naser Mahmood, Abdulmohsen Almalawi, Albert Y. Zomaya, Khalil Drira, A taxonomy of supervised learning for IDSs in SCADA environments, ACM Comput. Surv. 53 (2) (2020).

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

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

[44] Stefan Axelsson, Intrusion Detection Systems: A Survey and Taxonomy, Technical report, Technical report Chalmers University of Technology, Goteborg, Sweden, 2000.

[45] Tiranuch Anantvalee, Jie Wu, A survey on intrusion detection in mobile ad hoc networks, in: Wireless Network Security, Springer, 2007, pp. 159–180.

[46] You Chen, Yang Li, Xue-Qi Cheng, Li Guo, Survey and taxonomy of feature selection algorithms in intrusion detection system, in: Information Security and Cryptology, Springer, 2006, pp. 153–167.

[47] M. Alabadi, Z. Albayrak, Q-learning for securing cyber-physical systems : A survey, in: 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), IEEE, 2020, pp. 1–13.

[48] B. Zhu, S. Sastry, SCADA-specific intrusion detection / prevention systems : A survey and taxonomy, 2010.

[49] A.M. Aleesa, B.B. Zaidan, A.A. Zaidan, Nan M. Sahar, 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.

[50] Kelton A.P. da Costa, João P. Papa, Celso O. Lisboa, Roberto Munoz, Victor Hugo C. de Albuquerque, Internet of Things: A survey on machine learning-based intrusion detection approaches, Comput. Netw. 151 (2019) 147–157.

[51] Preeti Mishra, Vijay Varadharajan, Uday Tupakula, Emmanuel S. Pilli, A detailed investigation and analysis of using machine learning techniques for intrusion detection, IEEE Commun. Surv. Tutor. 21 (1) (2018) 686–728.

[52] Philipp Kreimel, Oliver Eigner, Paul Tavolato, Anomaly-based detection and classification of attacks in cyber-physical systems, in: Proceedings of the 12th International Conference on Availability, Reliability and Security, in: ARES '17, Association for Computing Machinery, New York, NY, USA, 2017.

[53] Prashanth Krishnamurthy, Ramesh Karri, Farshad Khorrami, Anomaly detection in real-time multi-threaded processes using hardware performance counters, IEEE Trans. Inf. Forensics Secur. 15 (2019) 666–680.

[54] Xueyang Wang, Charalambos Konstantinou, Michail Maniatakos, Ramesh Karri, Serena Lee, Patricia Robison, Paul Stergiou, Steve Kim, Malicious firmware detection with hardware performance counters, IEEE Trans. Multi-Scale Comput. Syst. 2 (3) (2016) 160–173.

[55] Patric Nader, Paul Honeine, Pierre Beauseroy, -Norms in one-class classification for intrusion detection in SCADA systems, Ind. Inform. IEEE Trans. 10 (4) (2014) 2308–2317.

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

[57] Peng Li, Oliver Niggemann, Non-convex hull based anomaly detection in CPPS, Eng. Appl. Artif. Intell. 87 (2020) 103301.

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

[59] Bin Zhang, Jia-Hai Yang, Jian-Ping Wu, Ying-Wu Zhu, Diagnosing traffic anomalies using a two-phase model, J. Comput. Sci. Tech. 27 (2) (2012) 313–327.

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

[61] S. Armina Foroutan, Farzad R. Salmasi, Detection of false data injection attacks against state estimation in smart grids based on a mixture Gaussian distribution learning method, IET Cyber-Phys. Syst.: Theory Appl. 2 (4) (2017) 161–171.

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

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

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

[65] Eirini Anthi, Lowri Williams, Małgorzata Słowińska, George Theodorakopoulos, Pete Burnap, A supervised intrusion detection system for smart home IoT devices, IEEE Internet Things J. 6 (5) (2019) 9042–9053.

[66] Saleh Soltan, Prateek Mittal, H. Vincent Poor, Line failure detection after a cyber-physical attack on the grid using bayesian regression, IEEE Trans. Power Syst. 34 (5) (2019) 3758–3768.

[67] Weizhong Yan, Lalit K. Mestha, Masoud Abbaszadeh, Attack detection for securing cyber physical systems, IEEE Internet Things J. 6 (5) (2019) 8471–8481.

[68] Chuadhry Mujeeb Ahmed, Martin Ochoa, Jianying Zhou, Aditya P. Mathur, Rizwan Qadeer, Carlos Murguia, Justin Ruths, NoisePrint: 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, in: ASIACCS '18, Association for Computing Machinery, New York, NY, USA, 2018, pp. 483–497.

[69] Hussam Amrouch, Prashanth Krishnamurthy, Naman Patel, Jörg Henkel, Ramesh Karri, Farshad Khorrami, 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, 2017, pp. 1–10.

[70] Hamid Reza Ghaeini, Nils Ole Tippenhauer, Jianying Zhou, Zero residual attacks on industrial control systems and stateful countermeasures, in: Proceedings of the 14th International Conference on Availability, Reliability and Security, in: ARES '19, Association for Computing Machinery, New York, NY, USA, 2019.

[71] Alexander N. Sokolov, Andrey N. Ragozin, Ilya A. Pyatnitsky, Sergei K. Alabugin, 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, in: SIN '19, Association for Computing Machinery, New York, NY, USA, 2019.

[72] Dimitrios Kosmanos, Apostolos Pappas, Leandros Maglaras, Sotiris Moschoyiannis, Francisco J Aparicio-Navarro, Antonios Argyriou, Helge Janicke, A novel intrusion detection system against spoofing attacks in connected electric vehicles, Array 5 (2020) 100013.

[73] V. Ariharan, Subha P. Eswaran, Srinivasarao Vempati, Naveed Anjum, Machine learning quorum decider (MLQD) for large scale IoT deployments, Procedia Comput. Sci. 151 (2019) 959–964.

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

[75] Ajay Kumara, C.D. Jaidhar, Automated multi-level malware detection system based on reconstructed semantic view of executables using machine learning techniques at VMM, Future Gener. Comput. Syst. 79 (2018) 431–446.

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

[77] Zhiwei Feng, Nan Guan, Mingsong Lv, Wenchen Liu, Qingxu Deng, Xue Liu, Wang Yi, Efficient drone hijacking detection using two-step GA-XGBoost, J. Syst. Archit. 103 (2020) 101694.

[78] Melissa Stockman, Dipankar Dwivedi, Reinhard Gentz, Sean Peisert, Detecting control system misbehavior by fingerprinting programmable logic controller functionality, Int. J. Crit. Infrastruct. Prot. 26 (2019) 100306.

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

[80] Sparsh Sharma, Ajay Kaul, Hybrid fuzzy multi-criteria decision making based multi cluster head dolphin swarm optimized IDS for VANET, Veh. Commun. 12 (2018) 23–38.

[81] Cosimo Ieracitano, Ahsan Adeel, Francesco Carlo Morabito, Amir Hussain, A novel statistical analysis and autoencoder driven intelligent intrusion detection approach, Neurocomputing 387 (2020) 51–62.

[82] M.R. Gauthama Raman, Nivethitha Somu, Sahruday Jagarapu, Tina Manghnani, Thirumaran Selvam, Kannan Krithivasan, V.S. Shankar Sriram, An efficient intrusion detection technique based on support vector machine and improved binary gravitational search algorithm, Artif. Intell. Rev. 53 (2019) 1–32.

[83] Waheed A.H.M. Ghanem, Aman Jantan, A new approach for intrusion detection system based on training multilayer perceptron by using enhanced Bat algorithm, Neural Comput. Appl. (2019) 1–34.

[84] Lida Haghnegahdar, Yong Wang, A whale optimization algorithm-trained artificial neural network for smart grid cyber intrusion detection, Neural Comput. Appl. (2019) 1–15.

[85] Yu-Qi Li, Li-Quan Xiao, Jing-Hua Feng, Bin Xu, Jian Zhang, AquaSee: Predict load and cooling system faults of supercomputers using chilled water data, J. Comput. Sci. Tech. 35 (1) (2020) 221–230.

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

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

[88] Safa Otoum, Burak Kantarci, Hussein T. Mouftah, On the feasibility of deep learning in sensor network intrusion detection, IEEE Netw. Lett. 1 (2) (2019) 68–71.

[89] Bo Zhang, Chunxia Dou, Dong Yue, Zhanqiang Zhang, Response hierarchical control strategy of communication data disturbance in micro-grid under the concept of cyber physical system, IET Gener. Transm. Distrib. 12 (21) (2018) 5867–5878.

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

[91] José M. Balbuena Palácios, Jorge R. Beingolea Garay, Alexadre M. Oliveira, Sergio T. Kofuji, Intrusion detection system: A hybrid approach for cyber-physical environments, Technology 39 (2013) 193–204.

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

[93] Jordan Landford, Rich Meier, Richard Barella, Xinghui Zhao, Eduardo Cotilla-Sanchez, Robert B. Bass, Scott Wallace, Fast sequence component analysis for attack detection in synchrophasor networks, 2015, arXiv preprint arXiv: 1509.05086.

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

[95] Raymond C. Borges Hink, Justin M. Beaver, Mark A. Buckner, Tommy Morris, Uttam Adhikari, Shengyi Pan, Machine learning for power system disturbance and cyber-attack discrimination, in: Resilient Control Systems (ISRCS), 2014 7th International Symposium on, IEEE, 2014, pp. 1–8.

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

[97] Justin M. Beaver, Raymond C. Borges-Hink, Mark A. Buckner, 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, 2013, pp. 54–59.

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

[99] Qin Lin, Sridha Adepu, Sicco Verwer, Aditya Mathur, 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, in: ASIACCS '18, Association for Computing Machinery, New York, NY, USA, 2018, pp. 525–536.

[100] Hongbo Liu, Yan Wang, Jian Liu, Jie Yang, Yingying Chen, Practical user authentication leveraging channel state information (CSI), in: Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security, in: ASIA CCS '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 389–400.

[101] Koyena Pal, Sridhar Adepu, Jonathan Goh, 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, 2017, pp. 124–127.

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

[103] Sudha Krishnamurthy, Soumik Sarkar, Ashutosh Tewari, 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, 2014, V002T26A006.

[104] Matti Mantere, Mirko Sailio, Sami Noponen, A module for anomaly detection in ICS networks, in: Proceedings of the 3rd International Conference on High Confidence Networked Systems, in: HiCoNS '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 49–56.

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

[106] Saeed Ahmed, YoungDoo Lee, Seung-Ho Hyun, Insoo Koo, Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forest, IEEE Trans. Inf. Forensics Secur. 14 (10) (2019) 2765–2777.

[107] Thiago Alves, Rishabh Das, Thomas Morris, Embedding encryption and machine learning intrusion prevention systems on programmable logic controllers, IEEE Embedded Syst. Lett. 10 (3) (2018) 99–102.

[108] Sudeep Pasricha, Janardhan Rao Doppa, Krishnendu Chakrabarty, Saideep Tiku, Daniel Dauwe, Shi Jin, Partha Pratim Pande, 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, 2017, pp. 1–10.

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

[110] Abdullah Khalili, Ashkan Sami, SysDetect: A systematic approach to critical state determination for Industrial Intrusion Detection Systems using Apriori algorithm, J. Process Control 32 (2015) 154–160.

[111] Dina Hadžiosmanović, Lorenzo Simionato, Damiano Bolzoni, Emmanuele Zambon, Sandro Etalle, 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, 2012, pp. 354–373.

[112] Sooyeon Shin, Taekyoung Kwon, Gil-Yong Jo, Youngman Park, Haekyu Rhy, An experimental study of hierarchical intrusion detection for wireless industrial sensor networks, Ind. Inform. IEEE Trans. 6 (4) (2010) 744–757.

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

[114] Y. Kwon, H.K. Kim, Y.H. Lim, J.I. Lim, A behavior-based intrusion detection technique for smart grid infrastructure, in: 2015 IEEE Eindhoven PowerTech, IEEE, 2015, pp. 1–6.

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

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

[117] Muhammad Azmi Umer, Aditya Mathur, Khurum Nazir Junejo, Sridhar Adepu, 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, in: CPS '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 131–136.

[118] Muhammad Azmi Umer, Aditya Mathur, Khurum Nazir Junejo, Sridhar Adepu, Generating invariants using design and data-centric approaches for distributed attack detection, Int. J. Crit. Infrastruct. Prot. 28 (2020) 100341.

[119] Chuadhry Mujeeb Ahmed, Muhammad Azmi Umer, Beebi Siti Salimah Binte Liyakkathali, Muhammad Taha Jilani, Jianying Zhou, Machine learning for CPS security: Applications, challenges and recommendations, in: Machine Intelligence and Big Data Analytics for Cybersecurity Applications, Springer, 2021, pp. 397–421.

[120] Shameek Bhattacharjee, Aditya Thakur, Sajal K. Das, 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, in: ASIACCS '18, Association for Computing Machinery, New York, NY, USA, 2018, pp. 173–185.

[121] Patrick Düssel, Christian Gehl, Pavel Laskov, Jens-Uwe Bußer, Christof Störmann, Jan Kästner, Cyber-critical infrastructure protection using real-time payload-based anomaly detection, in: Critical Information Infrastructures Security, Springer, 2009, pp. 85–97.

[122] Dayu Yang, Alexander Usynin, J. Wesley Hines, 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, 2006, pp. 12–16.

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

[124] Thomas Morris, Bradley Srivastava, Wei Gao, Kalyan Pavurapu, Ram Reddi, A control system testbed to validate critical infrastructure protection concepts, Int. J. Crit. Infrastruct. Prot. 4 (2) (2011) 88–103, adepu2016usingand.

[125] Steven Cheung, Bruno Dutertre, Martin Fong, Ulf Lindqvist, Keith Skinner, Alfonso Valdes, Using model-based intrusion detection for SCADA networks, in: Proceedings of the SCADA Security Scientific Symposium, Vol. 46, Citeseer, 2007, pp. 1–12.

[126] Robin Berthier, William H. Sanders, Specification-based intrusion detection for advanced metering infrastructures, in: Dependable Computing (PRDC), 2011 IEEE 17th Pacific Rim International Symposium on, IEEE, 2011, pp. 184–193.

[127] Raheel Shaikh, Feature selection techniques in machine learning with python, 2018, https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e.

[128] Anjum Nazir, Rizwan Ahmed Khan, A novel combinatorial optimization based feature selection method for network intrusion detection, Comput. Secur. 102 (2021) 102164.

[129] Vassilis G. Kaburlasos, Eleni Vrochidou, Fotios Panagiotopoulos, Charalampos Aitsidis, Alexander Jaki, Time series classification in cyber-physical system applications by intervals' numbers techniques, in: 2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), IEEE, 2019, pp. 1–6.

[130] Stephen D. Bay, Dennis Kibler, Michael J. Pazzani, Padhraic Smyth, The UCI KDD archive of large data sets for data mining research and experimentation, ACM SIGKDD Explor. Newsl. 2 (2) (2000) 81–85.

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

[132] iTrust, Dataset and Models, Center for Research in Cyber Security, Singapore University of Technology and Design, 2015, https://itrust.sutd.edu.sg/itrust-labs_datasets/dataset_info/#swat.

[133] Jun Inoue, Yoriyuki Yamagata, Yufi Chen, Christopher M. Poskitt, Jun Sun, 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, 2017, pp. 1058–1065.

[134] Khurum Nazir Junejo, Jonathan Goh, 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, in: CPSS '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 34–43.

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

[136] David Martin Powers, Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation, J. Mach. Learn. Technol. 2 (1) (2011) 37–63.

[137] Sean Whalen, Nathaniel Boggs, Salvatore J. Stolfo, Model aggregation for distributed content anomaly detection, in: Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop, in: AISec '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 61–71.

[138] Colin O'Reilly, Alexander Gluhak, Muhammad Ali Imran, Distributed anomaly detection using minimum volume elliptical principal component analysis, IEEE Trans. Knowl. Data Eng. 28 (9) (2016) 2320–2333.

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

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

[141] Bernardino Romera-Paredes, Philip H.S. Torr, An embarrassingly simple approach to zero-shot learning, in: Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37, in: ICML'15, JMLR.org, Lille, France, 2015, pp. 2152–2161.

[142] Richard Socher, Milind Ganjoo, Christopher D. Manning, Andrew Y. Ng, Zero-shot learning through cross-modal transfer, in: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1, in: NIPS'13, Curran Associates Inc., Red Hook, NY, USA, 2013, pp. 935–943.

[143] Iftikhar Ahmad, Muhammad Hussain, Abdullah Alghamdi, Abdulhameed Alelaiwi, Enhancing SVM performance in intrusion detection using optimal feature subset selection based on genetic principal components, Neural Comput. Appl. 24 (7–8) (2014) 1671–1682.

[144] Omar Al-Jarrah, Ahmad Arafat, Network intrusion detection system using neural network classification of attack behavior, J. Adv. Inf. Technol. 6 (1) (2015) 1–8.

[145] Z. Muda, W. Yassin, M.N. Sulaiman, N.I. Udzir, Intrusion detection based on K-means clustering and Naïve Bayes classification, in: Information Technology in Asia (CITA 11), 2011 7th International Conference on, IEEE, 2011, pp. 1–6.

[146] Vipin Kumar, Himadri Chauhan, Dheeraj Panwar, K-means clustering approach to analyze NSL-KDD intrusion detection dataset, Int. J. Soft Comput. Eng. (IJSCE) ISSN (2013) 2231–2307.

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

[148] Yichi Zhang, Lingfeng Wang, Weiqing Sun, Robert C Green, Mansoor Alam, et al., 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, 2011, pp. 1–8.

[149] Chih-Fong Tsai, Yu-Feng Hsu, Chia-Ying Lin, Wei-Yang Lin, Intrusion detection by machine learning: A review, Expert Syst. Appl. 36 (10) (2009) 11994–12000.

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

[151] J.R. Quinlan, Improved use of continuous attributes in C4.5, J. Artif. Int. Res. 4 (1) (1996) 77–90.

[152] Leo Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.

[153] Shailendra Sahu, B.M. Mehtre, Network intrusion detection system using J48 decision tree, in: Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on, IEEE, 2015, pp. 2023–2026.

[154] Md Al Mehedi Hasan, Mohammed Nasser, Biprodip Pal, Shamim Ahmad, Support vector machine and random forest modeling for intrusion detection systems, J. Intell. Learn. Syst. Appl. 2014 (2014) 45–52.

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

[156] Levent Koc, Thomas A. Mazzuchi, Shahram Sarkani, A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier, Expert Syst. Appl. 39 (18) (2012) 13492–13500.

[157] Liyuan Xiao, Yetian Chen, Carl K. Chang, Bayesian model averaging of Bayesian network classifiers for intrusion detection, in: Computer Software and Applications Conference Workshops (COMPSACW), 2014 IEEE 38th International, IEEE, 2014, pp. 128–133.

[158] Nir Friedman, Dan Geiger, Moises Goldszmidt, Bayesian network classifiers, Mach. Learn. 29 (2–3) (1997) 131–163.

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

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

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

[162] Tyler Giallanza, Travis Siems, Elena Smith, Erik Gabrielsen, Ian Johnson, Mitchell A. Thornton, Eric C. Larson, Keyboard snooping from mobile phone arrays with mixed convolutional and recurrent neural networks, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 3 (2) (2019) 1–22.

[163] James Le, The 10 deep learning methods AI practitioners need to apply, 2017, https://medium.com/cracking-the-data-science-interview/the-10-deep-learning-methods-ai-practitioners-need-to-apply-885259f402c1.

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

[165] Santiago Fernández, Alex Graves, Jürgen Schmidhuber, An application of recurrent neural networks to discriminative keyword spotting, in: International Conference on Artificial Neural Networks, Springer, 2007, pp. 220–229.

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

[167] Lior Rokach, Oded Maimon, Clustering methods, in: Data Mining and Knowledge Discovery Handbook, Springer, 2005, pp. 321–352.

[168] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, 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, in: KDD'96, AAAI Press, 1996, pp. 226–231.

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

[170] Aymen Abid, Abdennaceur Kachouri, Adel Mahfoudhi, Outlier detection for wireless sensor networks using density-based clustering approach, IET Wirel. Sensor Syst. 7 (4) (2017) 83–90.

[171] Charu C. Aggarwal, Chandan K. Reddy, Data Clustering: Algorithms and Applications, first ed., Chapman & Hall/CRC, 2013.

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

[173] Jungsuk Song, Kenji Ohira, Hiroki Takakura, Yasuo Okabe, Yongjin Kwon, A clustering method for improving performance of anomaly-based intrusion detection system, IEICE Trans. Inf. Syst. 91 (5) (2008) 1282–1291.

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

[175] Y. Guan, A.A. Ghorbani, N. Belacel, 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, 2003, pp. 1083–1086.

[176] James MacQueen, et al., 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, 1967, pp. 281–297.

[177] Kun-Lun Li, Hou-Kuan Huang, Sheng-Feng Tian, Wei Xu, 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, 2003, pp. 3077–3081.

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

[179] Rakesh Agrawal, Tomasz Imieliński, Arun Swami, Mining association rules between sets of items in large databases, SIGMOD Rec. 22 (2) (1993) 207–216.

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

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

[182] Bo Yang, Xiao Fu, Nicholas D. Sidiropoulos, Mingyi Hong, Towards K-means-friendly spaces: Simultaneous deep learning and clustering, in: Proceedings of the 34th International Conference on Machine Learning - Volume 70, in: ICML'17, JMLR.org, 2017, pp. 3861–3870.

[183] Peihao Huang, Yan Huang, Wei Wang, Liang Wang, Deep embedding network for clustering, in: 2014 22nd International Conference on Pattern Recognition, IEEE, 2014, pp. 1532–1537.

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

[185] Kamran Ghasedi Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, Heng Huang, Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE, 2017, pp. 5736–5745.

[186] Sohil Atul Shah, Vladlen Koltun, Deep continuous clustering, 2018, arXiv preprint arXiv:1803.01449.

[187] Léon Bottou, Large-scale machine learning with stochastic gradient descent, in: Proceedings of COMPSTAT'2010, Springer, 2010, pp. 177–186.

[188] Robert Hecht-Nielsen, 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: Harry Wechsler (Ed.), Neural Networks for Perception, Academic Press, 1992, pp. 65–93.

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

[190] Jost Tobias Springenberg, Unsupervised and semi-supervised learning with categorical generative adversarial networks, 2015, arXiv preprint arXiv:1511.06390.

[191] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, Pieter Abbeel, InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, in: NIPS'16, Curran Associates Inc., Red Hook, NY, USA, 2016, pp. 2180–2188.

[192] Antonio Criminisi, Jamie Shotton, Ender Konukoglu, Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning, Found. Trends Comput. Graph. Vis. 7 (2–3) (2012) 81–227.

[193] Khurum Nazir Junejo, Asim Karim, Robust personalizable spam filtering via local and global discrimination modeling, Knowl. Inf. Syst. 34 (2) (2013) 299–334.

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

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

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

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

[198] Christopher T. Symons, Justin M. Beaver, 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, in: AISec '12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 49–58.

[199] Sharmila Wagh, Anagha Khati, Auzita Irani, Naba Inamdar, Rashmi Soni, Effective framework of J48 algorithm using semi-supervised approach for intrusion detection, Int. J. Comput. Appl. 94 (12) (2014).

[200] Guohong Gao, Guoyi Miao, Jiaxia Sun, Yafeng Han, Improved semi-supervised fuzzy clustering algorithm and application in effective intrusion detection system, Int. J. Adv. Comput. Technol. 5 (4) (2013).

[201] R.S. Sutton, A.G. Barto, Introduction to Reinforcement Learning, MIT Press, Cambridge, MA, 1998.

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

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

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

[205] Martina Panfili, Alessandro Giuseppi, Andrea Fiaschetti, Homoud B. Al-Jibreen, Antonio Pietrabissa, Franchisco Delli Priscoli, 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, 2018, pp. 460–465.

[206] AUBIGNY, A. Consortium, "ATENA website", 2017, https://www.atena-h2020.eu/.

[207] Jun Yan, Haibo He, Xiangnan Zhong, Yufei Tang, Q-learning-based vulnerability analysis of smart grid against sequential topology attacks, IEEE Trans. Inf. Forensics Secur. 12 (1) (2016) 200–210.

[208] Huimin Lu, Yujie Li, Shenglin Mu, Dong Wang, Hyoungseop Kim, Seiichi Serikawa, Motor anomaly detection for unmanned aerial vehicles using reinforcement learning, IEEE Internet Things J. 5 (4) (2017) 2315–2322.

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

[210] Junyu Lin, Lei Xu, Yingqi Liu, Xiangyu Zhang, 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, 2020, pp. 113–131.

[211] Jiaheng Zhang, Zhiyong Fang, Yupeng Zhang, Dawn Song, Zero knowledge proofs for decision tree predictions and accuracy, in: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, 2020, pp. 2039–2053.

[212] Yu Li, Min Li, Bo Luo, Ye Tian, Qiang Xu, DeepDyve: Dynamic verification for deep neural networks, in: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, 2020, pp. 101–112.

[213] Muhammad Azmi Umer, Chuadhry Mujeeb Ahmed, Muhammad Taha Jilani, Aditya P. Mathur, Attack rules: an adversarial approach to generate attacks for Industrial Control Systems using machine learning, in: Proceedings of the 2th Workshop on CPS&IoT Security and Privacy, 2021, pp. 35–40.

[214] Gayathri Sugumar, Aditya Mathur, A method for testing distributed anomaly detectors, Int. J. Crit. Infrastruct. Prot. 27 (2019) 100324.

[215] A.P. Mathur, N.O. Tippenhauer, SWaT: a water treatment testbed for research and training on ICS security, in: 2016 International Workshop on Cyber-Physical Systems for Smart Water Networks (CySWater), 2016, pp. 31–36.

[216] Yifan Jia, Jingyi Wang, Christopher M. Poskitt, Sudipta Chattopadhyay, Jun Sun, Yuqi Chen, Adversarial attacks and mitigation for anomaly detectors of cyber-physical systems, Int. J. Crit. Infrastruct. Prot. 34 (2021) 100452.

[217] Chuadhry Mujeeb Ahmed, Venkata Reddy Palleti, Aditya P. Mathur, WADI: A water distribution testbed for research in the design of secure cyber physical systems, in: CysWater, ACM, NY, USA, 2017.

[218] Yixin Sun, Kangkook Jee, Suphannee Sivakorn, Zhichun Li, Cristian Lumezanu, Lauri Korts-Parn, Zhenyu Wu, Junghwan Rhee, Chung Hwan Kim, Mung Chiang, et al., Detecting malware injection with program-DNS behavior, in: 2020 IEEE European Symposium on Security and Privacy (EuroS&P), IEEE, 2020, pp. 552–568.

[219] P. Mishra, P. Aggarwal, A. Vidyarthi, P. Singh, B. Khan, H. Haes Alhelou, P. Siano, VMShield: Memory introspection-based malware detection to secure cloud-based services against stealthy attacks, IEEE Trans. Ind. Inf. (2021) 1.

[220] Zhai Liang, Tang Xinming, Li Lin, Jiang Wenliang, 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, 2005.