A Survey on Representation Learning Efforts in Cybersecurity Domain

MUHAMMAD USMAN, Swinburne University of Technology, Australia
MIAN AHMAD JAN, Abdul Wali Khan University Mardan, Pakistan
XIANGJIAN HE, University of Technology Sydney, Australia
JINJUN CHEN, Swinburne University of Technology, Australia

In this technology-based era, network-based systems are facing new cyber-attacks on daily bases. Traditional cybersecurity approaches are based on old threat-knowledge databases and need to be updated on a daily basis to stand against new generation of cyber-threats and protect underlying network-based systems. Along with updating threat-knowledge databases, there is a need for proper management and processing of data generated by sensitive real-time applications. In recent years, various computing platforms based on representation learning algorithms have emerged as a useful resource to manage and exploit the generated data to extract meaningful information. If these platforms are properly utilized, strong cybersecurity systems can be developed to protect the underlying network-based systems and support sensitive real-time applications. In this survey, we highlight various cyber-threats, real-life examples, and initiatives taken by various international organizations. We discuss various computing platforms based on representation learning algorithms to process and analyze the generated data. We highlight various popular datasets introduced by well-known global organizations that can be used to train the representation learning algorithms to predict and detect threats. We also provide an in-depth analysis of research efforts based on representation learning algorithms made in recent years to protect the underlying network-based systems against current cyber-threats. Finally, we highlight various limitations and challenges in these efforts and available datasets that need to be considered when using them to build cybersecurity systems.

CCS Concepts: • **Networks** → Network properties; • **Security and privacy** → Access control; • **Computing methodologies** → Machine learning.

Additional Key Words and Phrases: cyber-attacks, cybersecurity, computing, representation learning, datasets

ACM Reference Format:
Muhammad Usman, Mian Ahmad Jan, Xiangjian He, and Jinjun Chen. 2019. A Survey on Representation Learning Efforts in Cybersecurity Domain. 1, 1 (March 2019), 27 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In this Internet-based era, millions of devices are connected to the Internet. These devices help in enhancing the performance of various applications by sharing various computational and

This paper's research is partially supported by Australian Research Council projects of DP170100136 and LP140100816.
Authors' addresses: Muhammad Usman, Swinburne University of Technology, Hawthorn Campus, 3122, Melbourne, Victoria, Australia, muhammad.usmanskk@gmail.com; Mian Ahmad Jan, Abdul Wali Khan University Mardan, Garden Campus, 23200, Mardan, Khyber Pakhtunkhwa, Pakistan, mianjan@awkum.edu.pk; Xiangjian He, University of Technology Sydney, Broadway Campus, 2007, Sydney, New South Wales, Australia, xiangjian.he@uts.edu.au; Jinjun Chen, Swinburne University of Technology, Hawthorn Campus, 3122, Melbourne, Victoria, Australia, jinjun.chen@gmail.com.

XXXX-XXXX/2019/3-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

storage resources. These devices and the connections between them need to be protected using various cybersecurity technologies. Physical devices and cybersecurity technologies together build systems known as cybersecurity systems [41, 99, 119]. These systems are used to support various critical applications, such as transportation management, healthcare, surveillance systems, and environmental monitoring, in environments like smart cities. These applications operate in real-time and generate big sensitive data. Furthermore, these applications need to be protected from internal and external threats [5, 36, 46, 54, 118]. To protect these applications and the generated data, there is a need for strong cybersecurity systems that can analyze the generated data in real-time. The cybersecurity systems need to support different types of connections, devices, and applications. These systems need to be updated automatically and provide protection to the underlying network-based systems and applications running on them from cyber-threats.

In the traditional network-based systems, multiple cybersecurity applications are used together to protect connected devices from various threats [43, 103]. These applications help in detecting and identifying attacks in various forms, e.g., unauthorized access, viruses, and privacy leakage. Furthermore, they save the underlying network-based systems from possible physical destruction and manipulation of data. The cyber-attacks can be internal or external. The cybersecurity applications detect and identify these attacks through known signatures or by analyzing the behaviors of the underlying systems. In the signature-based detection category, knowledge-based databases are required. These databases need to be updated manually. As a result, the cybersecurity applications falling in this category are not feasible for real-time applications and systems [87, 108]. In the behavior analysis category, the behaviors of underlying systems are analyzed continuously to differentiate between normal and abnormal activities [7, 96]. This category is useful for real-time systems, however, it has a very high ratio of false alarms and may disrupt the performances of underlying systems. Limitations in existing signature-based detection and behavior analysis categories make the job of attackers and intruders easy and they can enter the systems through hidden doors. Therefore, there is a need to analyze and study existing and newly developed cybersecurity systems and applications to highlight their weaknesses and suggest possible improvements by incorporating machine learning algorithms.

In the recent years, many researchers have started using machine learning algorithms in the cybersecurity domain to train Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) [2, 92]. To accurately identify various cyber-attacks, the behaviors of the underlying systems and the generated data need to be analyzed carefully. In real-time environments, the network-based systems generate huge volumes of data. To process and analyze the generated big data, different machine learning and data mining algorithms can be utilized to detect anomalies, identify threats, and classify familiar and unfamiliar entities as shown in Fig. 1. The machine learning and data mining algorithms can further be classified into various subcategories, out of which the most popular one is representation learning algorithms. The representation learning algorithms allow a system to automatically discover representations required to detect or classify features from raw input data. This category consists of well-known machine learning algorithms from supervised, unsupervised, and deep architectures to support a diverse range of applications. If these algorithms are utilized properly, new intelligent IDS and IPS can be designed to make intelligent and quick decisions with high accuracy levels by learning from real-time data generated by network-based systems at low computational costs.

In this survey, we focus on the use of representation learning algorithms in the cybersecurity domain. Many survey papers on machine learning algorithms can be found in the literature [2, 13, 89]. However, these papers discuss machine learning algorithms in general and there is no specific survey on representation learning algorithms for cybersecurity applications. In our survey paper, we review literature on various representation learning algorithms. To the best of our

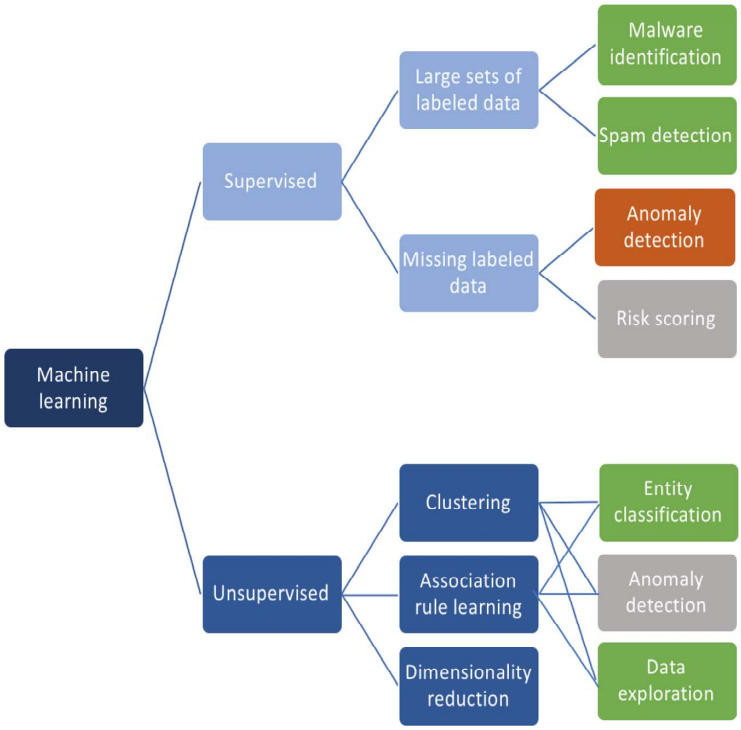

Fig. 1. Machine learning algorithms and threat detection

knowledge, this paper is the first survey on representation learning algorithms for cybersecurity applications. Our intention is to provide an overview of representation learning algorithms that can be used to process generated data for various purposes in the cybersecurity domain. There are multiple subdomains in cybersecurity, e.g., security, privacy, forensics, etc. In this article, we narrow down our literature and target security domain only. We discuss well-known cyber-attacks and collaborations between various international organizations. We review popular computing platforms based on representation learning algorithms. These computing platforms are managed by well-known vendors. We discuss various publicly available datasets designed for machine learning based cybersecurity systems. We also discuss most recent advancements based on representation learning algorithms for cybersecurity systems. In the end, we highlight limitations and research challenges when using the representation learning algorithms and publicly available datasets to design cybersecurity systems for real-time applications along with possible future research directions. The main contributions of this survey are listed below.

- We present and discuss various well-known cyber-threats. We provide examples of cyber-threats reported in the past few years. Various Initiatives taken by global organizations are also discussed in detail.
- We provide an in-depth overview of various computing platforms based on representation learning algorithms. These platforms are designed to process generated big data. These platforms help researchers from cybersecurity community to put minimum efforts to process and analyze the generated data.

- We provide a detailed discussion on various cybersecurity datasets. These datasets are introduced by well-known international organizations and are publicly available. The datasets are open source and can be used with different machine learning algorithms for further research and development. Limitations and research challenges of these datasets are also highlighted along with possible future research directions.
- Finally, we discuss various research efforts based on representation learning algorithms made in the recent years for cybersecurity applications. We highlight their limitations and research challenges when using them to design complex cybersecurity systems. We also highlight various possible future research directions.

The rest of this survey is organized as follows. Popular cyber-attacks and initiatives taken by global organizations are presented in Section 2. Various computing platforms based on representation learning algorithms to process generated big data are presented in Section 3. Section 4 provides a summary of publicly available datasets and research efforts based on representation learning algorithms for cybersecurity systems. Limitations, research challenges, and possible future research directions are presented in Section 5. Finally, the survey is concluded in Section 6.

2 CYBER-ATTACKS AND INITIATIVES

Cybersecurity techniques are considered as a trade-off between defenders and attackers [101]. The defenders need to be well-prepared against every possible attack from attackers. In simple words, they need to follow the proactive management policy [73]. On the other hand, the attackers keep an eye on the target, search for a weak point or a loophole to penetrate the targeted system. In the past few years, improvements have been seen in the defensive mechanisms that are based on agreements and coordination between various technologies [17, 52, 58, 82]. The attackers make use of the same technologies used by defenders to penetrate the targeted system for malicious purposes. However, the latest defensive mechanisms are very complex and difficult to manage by defenders. This has made the job of attackers quite easy. They can easily enter the targeted system through a weak spot. Once entered, the attackers can try new types of attacks (an advantage). On the other hand, the defenders need to be always alert and consistent in their defensive mechanisms (a disadvantage).

2.1 Cyber-Attack Categories

We can divide cyber-attacks into three broad categories, i.e., multi-vector, multi-stage, and hybrid (a combination of multi-vector and multi-stage categories). In the multi-vector category, different means of propagation, e.g., spam emails, online ads, and malware websites, are used to attack targeted systems [14]. This category requires connectivity to the Internet. In the multi-stage category, network traffic is usually monitored to plan an attack. Reconnaissance attack is a very common example in this category [23]. Current cyber-attacks mostly belong to the hybrid category and follow a kill-chain strategy. In this strategy, the attackers first apply the multi-stage category attacks to infiltrate the targeted systems. Once the attacks are successfully done, the attackers apply the multi-vector category in online or offline mode to exfiltrate network data and configurations of the underlying systems. Nowadays, the attackers apply various strategies, e.g., social engineering and polymorphic attacks, to authenticate themselves to bypass firewalls and spam filters. In the hybrid category, the attackers can easily evade the traditional security systems through step-by-step independent events without letting defenders know that a cyber-attack is about to happen. Some well-known cyber-attacks are discussed in the following subsections.

2.1.1 Polymorphic Attacks. The polymorphic attacks are a type of attacks in which the code of a malicious program changes itself each time when it runs, but the function of the code (its semantics)

does not change at all. The polymorphic attacks spread from one device to another device and travel in a network. This category of attacks includes worms, viruses, and Trojans [31]. Real-life examples in the recent years include The Storm worm (2007) [51] and Virlock ransomware family (2014) [97]. The main purpose of these attacks is to evade cybersecurity systems and introduce different effects, e.g., hiding files, changing file names, and corrupting root directories. Once entered the systems, the attacking applications create multiple instances with the same purpose. However, the payload of each instance can have a different code and if detected, the cybersecurity systems may not reliably predict the next attack. The polymorphic attacks make the instances look different from each other to successfully blended in the network traffic. As a result, the attacks can easily bypass the cybersecurity systems that rely on signatures or payload-based statistics. There are different vendors that create and distribute various signatures-based attacks to sell their security products in the consumer market. As a result, clients constantly purchase their products and become a part of a cycle which is beneficial to them and the attackers.

2.1.2 Persistent Attacks. The persistent attacks, also known as advanced persistent threats, consist of continuous and stealthy computer hacking processes to target a specific entity or an entire system [18]. Real-life examples in the recent years include the Titan Rain (2003) [34], Sykipot Attacks (2006) [112], GhostNet (2009) [104], Stuxnet Worm (2010) [61], and Deep Panda (2015) [39]. These attacks usually target private and public organizations and states with political and financial motives. These attacks rely on malware to exploit weak points of the targeted system. The main motive of these types of attacks is to monitor the activities or extract sensitive information from the targeted systems. Techniques like social engineering, supply-chain compromises, and infected media can also be used in these attacks. In the persistent attacks, a malicious code is usually placed in one or two computers of an organization for a long period to secretly monitor organizational activities, store sensitive information, and transmit the collected information to an intended destination. This category of attacks just monitors the targeted systems and does not destroy them. Their main targets are always specific organizations rather than individuals.

2.1.3 Composite Attacks. In the cybersecurity domain, two most popular attacks are syntactic and semantic attacks. The composite attacks are formed by combining these two types of attacks. The syntactic attacks are straightforward and operate at the software level. These attacks exploit technical vulnerabilities of a software to steal targeted data. Examples include viruses, worms, and Trojan horses [20]. Real-life examples of syntactic attacks in the recent years include CryptoLocker (2013) [71], Stuxnet Worm (2010) [61], and Deep Panda (2015) [39]. On the other hand, the semantic attacks try to modify correct information to set someone in a wrong direction [48]. Examples include phishing, application and file masquerading, false pop-ups, malware advertisements on websites, game and friend requests on social websites, and attacks through removable media [47]. In the recent years, the composite attacks are getting popularity. The attackers use social engineering techniques to gain access to privileged information and then apply an attack to bring harm to the targeted network or device. A well-known example of composite attacks is the phishing attack, also known as online scams [121].

2.1.4 Zero-Day Attacks. This is another software level attack and usually unknown to those (e.g., vendors) who mitigate this attack. It is named zero-day because this attack is launched before a vendor gets aware [63]. This type of attack is very effective because it can go undetectable for a long period, e.g., weeks or months. Even after detection, days or weeks are taken to eliminate its effects. Until the vulnerability is detected, the attackers can exploit the targeted systems or applications. This attack can be applied in different forms, such as worms and viruses. Real-life examples of

zero-day attacks in the recent years include EternalBlue (2017) and WannaCry ransomware (2017) [60].

2.2 Initiatives and Cooperation

There are many international organizations that share information about various cyber-attacks for researchers and developers. Some organizations just focus on incidents and vulnerability reports while others focus on intrusion detection and prevention. Together, these organizations play an important role in spreading awareness about various threats and propose implementable solutions. Some well-known organizations are discussed in the following subsections.

2.2.1 Computer Emerging Response Team. In The Computer Emerging Response Team (CERT), experts deal with incidents related to computer security [45]. This team is also known by other names, e.g., security incident response group or emergency readiness group. The CERT works on a regional level and collects information about cyber-attacks from various sources. After collecting the information, it can issue early warnings and provide help when requested. Due to its regional support, there are many groups linked to this team and are working together or independently in different countries. A cooperation between these groups on a global level is required to spread the awareness about various cyber-attacks and the available solutions.

2.2.2 Forum for Incident Response and Security Team. The Forum for Incident Response and Security Team (FIRST) is a globally recognized team. It is responsible to take quick actions on reported incidents in the cybersecurity domain. Member countries forward security incidents to different incident response teams of FIRST for further actions. The FIRST provides a platform to various governmental, educational, and commercial organizations so that they can share services of their computer security incident response teams. The main purpose is to foster coordination and cooperation for incident prevention, rapidly react to the reported incidents, and share reported information among member countries. Currently, more than four hundred regions from different countries in different continents, e.g., Africa, America, Asia, and Europe, hold the membership of the FIRST [90].

2.2.3 European Union Network and Information Security Agency. The European Union Network and Information Security Agency (ENISA) is a center to provide cybersecurity services to European countries [91]. It prepares and equips European countries to detect, prevent, and respond to various information security problems. It provides practical advice and solutions to its member European countries. It also helps in organizing different exercises related to cyber-crises, developing national cybersecurity strategies, and promoting cooperation between the CERTs. It is also involved in publishing reports and studies on various cybersecurity issues in various domains, e.g., cloud security, ensuring privacy in new networking technologies, electronic trust services, and identifying cyber-attacks.

2.2.4 Computer Security Division. The Computer Security Division (CDS) is one of the seven technical divisions in the information technology laboratory at the national institute of standards and technology [53]. It develops tests for standards, metrics, and guidelines to protect federal information systems. The standards and tools are developed in transparent and collaborative ways by involving experts around the world. The developed standards can voluntarily be adopted by various organizations due to their global acceptance. The main purpose of the developed standards is to deal with the present and future challenges related to information security. These standards provide methodologies to develop practical security applications and technologies. In 2015, this division was divided into two subdivisions, where the first division deals with computer security

and the second division deals with applied cybersecurity problems. These two subdivisions work closely together on various programs and projects.

3 REPRESENTATION LEARNING AND COMPUTING PLATFORMS

There are many algorithms in the machine learning domain that can be used for various purposes. In this domain, there is a set of algorithms that allows an intelligent system to analyze input data, automatically discover representations present in the input data, and use the discovered representations to apply different operations on the input data, e.g., data classification and feature extraction. As a result, the systems can become smart and operate without any assistance and manual feature extraction. Such an intelligence is required by systems that are analyzing real-time big data generated by underlying network-based systems. The input data can be a combination of multimedia and non-multimedia data. The analysis of mixed data requires mathematically and computationally complex processes. In traditional systems based on machine learning algorithms, the classification of data requires an input that should be easy to analyze and process. However, multimedia data, e.g., videos, do not contain a defined feature set. In this case, various representations and examinations are utilized to discover the required features. The representation learning algorithms can be used to infer specific patterns from input multimedia data. The inferred patterns can later be used to train computer programs and applications designed for specific tasks. Just like traditional machine learning algorithms, the representation learning algorithms can also be classified into three major categories, i.e., supervised, unsupervised and multilayer/deep architectures [50, 67, 74].

The supervised learning category requires a training dataset to make a final decision. In the first stage, the input and the expected output datasets are provided. Later, specific patterns are inferred from provided datasets to make a final decision. The same inferred patterns can be applied on newly arrived input and output datasets. In the unsupervised learning category, the algorithms do not use labeled training data. Algorithms falling in this category try to find hidden data patterns from input data without any guidance. The Multilayer/deep architectures are inspired from human neurological systems and stack multiple layers of learning nodes. These architectures are based on distributed representations. Multiple interactions are performed between different levels based on different factors to generate the observed data. Each intermediate layer generates an output for corresponding input data. The output of each level is used as an input for next level to produce a new representation, i.e., the original data are the input of the first level and final representations of features are the output of the last level. The algorithms in aforementioned categories and relevant literature are discussed in detail in Section 4.2.

3.1 Computing Platforms for Representation Learning

In cybersecurity systems, the generated data can be multimedia, non-multimedia, or a combination of both types. Manual processing and classification of such type of data can be a time-consuming task if the generated data are huge in size. To deal with this problem, there is a need for computing platforms to quickly process and classify the generated big data to make quick decisions. In the recent years, many Application Programming Interfaces (APIs) are introduced in the market to process big multimedia and non-multimedia data. These APIs are introduced by many well-known companies, such as Amazon, IBM, Microsoft, and Google. These APIs are general purpose and can be used to process different types of data generated by different applications. Few example applications are image processing, computer vision, natural language processing, signal processing, and cybersecurity. These APIs are based on popular machine learning algorithms and provide a platform to develop applications in a user-friendly way. These APIs hide usual complexities involved in developing machine learning algorithms. Therefore, the developers only need to focus on data manipulation and experimentation to design and deliver the required products. The main

purpose of this section is to inform readers and developers in the cybersecurity domain about various well-known APIs that can be used when designing cybersecurity systems based on machine or representation learning algorithms. These APIs offer simple interfaces, instructions, and easy manuals so that the developers can use them without having a strong background in the machine or representation learning domain. Furthermore, online help is also available if the developers are stuck at some point when using any of these tools. These APIs are explained in the following subsections. Furthermore, a comparison between these APIs in terms of capabilities and support is provided in Table 1.

Features	IBM Watson	Microsoft Azure	Google Prediction	Amazon	BigML
Support for different data sources	Advanced	Advanced	No support	No support	Advanced
Data transformation and cleaning capabilities	Advanced	Advanced	Basic	Basic	Advanced
Support for different machine learning algorithms	Advanced	Advanced	Advanced	Basic	Advanced
Support for creation of algorithms	Advanced	Advanced	No support	No support	Advanced
Support for algorithm performance evaluation	Advanced	Advanced	No support	Advanced	Advanced
Support for computed result evaluation	Advanced	Advanced	Advanced	No support	Advanced
Support for parameter tuning	Advanced	Advanced	Basic	Basic	Advanced

Table 1. Supported features of different computing platforms

3.1.1 ***IBM Watson API.*** This API helps in simplifying data preparation and analysis processes. Results of data analysis can be displayed by various visual tools. It is a free API and available for public use through IBM's Bluemix cloud service platform [109]. It helps developers to develop applications, services, and products with cognitive skills. The IBM Watson platform offers more than twenty-five APIs powered by fifty different technologies. Some of the features offered by this API are listed below.

- It can interpret text in different language pairs using machine translation.
- It can estimate popularity of a specific word or phrase using message resonance.
- It can provide answers to questions triggered by primary document resources.
- It can predict social characteristics of human beings from a given text.

3.1.2 ***Microsoft Azure API.*** This API is designed for data scientists and help them to perform a quick data analysis. It also helps them by saving their time from developing complex representation learning models. This API can be used to analyze multimedia data collected from diverse resources. With the help of predictive models, various abnormal events in different scenarios can be predicted using this API. It is available in various Microsoft products, e.g., Xbox and Bing, and offers strong representation learning abilities [9]. Some of the strong features offered by this API are listed below.

- By using this API, the data scientists can develop customized and configurable models to train and predict tasks based on their own R language code.
- By using various Python libraries, e.g., SciPy, Pandas and NumPy, the data scientists can include Python language scripts in this API. It also supports other popular Python tools, e.g., iPython Notebook and Visual Studio-based tools.
- By using this API, the data scientists can train petabytes of data using Principal Component Analysis (PCA) or Support Vector Machines (SVMs) to predict malicious behavior in the multimedia data.
- It supports other data processing platforms, e.g., Hadoop and Spark, to process huge volumes of big multimedia data.

3.1.3 ***Google Prediction API.*** This API is useful for real-time applications to quickly process collected big multimedia data. It is a cloud computing based facility and can be used in many applications, e.g., behavior analysis, spam detection, data classification, and event prediction [114]. By using this API, users can crunch big multimedia data to predict results without having strong programming skills and knowledge of machine learning algorithms. While using this API, the multimedia data need to be uploaded to Google cloud storage. Once the uploading is done, the data are read using BigQuery. This API can be used in many applications of Internet of Things (IoT) generating multimedia data, e.g., transportation management, surveillance, and healthcare. Two popular examples are listed below.

- The Ford is a company that designs and sells vehicles. This API is used in the Ford's laboratories for research purposes. The main focus is on how to improve driving skills of their clients by showing them maps of their daily routine routes. It also facilitates its clients to save routes and location information. Once saved, the vehicle becomes smart vehicles and can determine automatically where the driver wants to go. This decision can be made based on driver's driving routines at specific times and days of weeks. It can also help in analyzing the behavior of drivers during driving [72].
- The Pondera Solutions is a USA-based company to detect frauds. It uses the Google Prediction API to address various government issues, e.g., fraud, abuse, and waste in different public sectors [35].

3.1.4 ***Amazon API.*** It is another popular API to simplify thee processes of predictions, model building, data filtering, and statistical analysis [85]. Machine learning models offered by the Amazon deal with prediction problems only. Furthermore, the ratio of prediction error controls the speed of processing big multimedia data. Many visualization tools are offered by this API to facilitate its users to get a better insight into the processed data. However, there are certain restrictions in terms of user interface and representation learning algorithms. Nevertheless, it is user-friendly and easy to use API, and can be used for various tasks in IoT architectures, e.g., surveillance, transportation management, and healthcare. Some examples are listed below.

- It can help to find genre of songs by analyzing features of signal levels.
- It can help in recognizing human actions and activities from multimedia data. It can use geo-location information from non-multimedia data to predict users' activities.
- It can help to predict modes of payment (e.g., cash or card) by analyzing payments made in the first week in any shopping center.
- It can help in detecting fake users and identities by analyzing web activities.

3.1.5 ***BigML.*** It is a user-friendly API and based on decision trees. It helps representation learning developers by making predictive analytics tasks easy and understandable. It also helps in understanding business requirements and analysis reports. This API offers three modes of operations,

i.e., RESTful, web interface, and command line interface, out of which the web interface is the most popular one [8]. This API also supports real-time processing and data analysis. Some well-known examples are listed below.

- It helps in finding a relationship between different attributes of input multimedia data. It also helps in predicting features of similar objects in the collected multimedia data. The obtained results can help users to efficiently utilize specific objects.
- It helps to build predictive models based on some past examples from similar datasets. There is a support for batch processing jobs, i.e., designed models can process various data instances in a batch to save the computational time. However, the accuracy of prediction depends on the amount of provided data.
- It supports remote access to developed predictive models via the command line interface.

4 REPRESENTATION LEARNING AND CYBERSECURITY

There are two main phases in representation learning based systems, i.e., training and testing. But there can be a third phase called validation. The training phase is usually a time consuming process, and the purpose is to identify features and attributes of different classes and determine total number of classes from input data. Training datasets consist of various examples. These examples are used to fit parameters of models based on representation learning algorithms. This phase is required for supervised learning category. Once the training phase is completed, the next phase is to validate the performance by predicting responses. This phase is based on validation datasets. In the validation phase, the performance of the developed model is regularized through early stopping. During this phase, the training stops with an increase in the error in the validation datasets. Once the model is successfully passed through the validation phase, an unbiased evaluation is performed on it during the testing phase by using test datasets. The test datasets are independent of the training datasets and consist of different examples. These different examples are used to test the performance of the designed model. Hence, the datasets play an important role in the designing of representation learning models for specific tasks. In the following subsections, we discuss various packet capturing tools, datasets, and representation learning algorithms to perform various tasks in the cybersecurity domain.

4.1 Datasets

The datasets play an important role in representation learning algorithms. It is very important to have an understanding of datasets that can be used by different representation learning algorithms to perform various tasks in the cybersecurity domain. In the following subsections, we discuss a few datasets designed by well-known organizations for research purposes in the cybersecurity domain. These datasets are summarized in Table 2.

4.1.1 ***DARPA Dataset.*** The Defense Advanced Research Projects Agency (DARPA) is a well-known agency from the United States department of defense. Its main purpose is to focus on developing technologies for military purposes [113]. In the DARPA, there is a formal intrusion detection evaluation group, known as Cyber Systems and Technology Group (CSTG) of MIT Lincoln Laboratory that works under the sponsorship of defense advanced research projects agency and air force research laboratory. To evaluate a cybersecurity system designed for computer networks, the CSTG made evaluation efforts in the years 1998 and 1999. The main purpose of these efforts is to estimate rates of true and false alarms for all under evaluation systems. The evaluations are designed to be simple and the main focus is on core technology issues. Another motive of these evaluations is to provide data types for cybersecurity systems. There are many offline datasets available that can be used by researchers. These datasets contain various examples of background

Datasets	Vendor	Features
DARPA	Cyber systems and technology group	• Evaluation based on network traffic and audit logs, • support for batch mode processing, • support for Windows and Linux operating systems, • can detect different types of attacks
ADFA	Australian defence force academy	• Evaluation based on network traffic and audit logs, • support for Windows and Linux operating systems, • can detect different types of attacks
NetReSec	Network forensics and network security monitoring	• Evaluation based on pcap files, • support for Windows and Linux operating systems, • can detect different types of attacks
CRC	Cyber research center	• Evaluation based on pcap and log files, • support for Windows and Linux operating systems, • can detect different types of attacks

Table 2. Well-known cybersecurity datasets

data traffic and possible attacks. Two datasets were created from DARPA's intrusion detection evaluations in the years 1998 and 1999. Both datasets offer offline and real-time evaluations. In the offline evaluation, the network traffic and audit logs are processed in a batch mode to identify attacks during usual network activities. In the real-time evaluation, the cybersecurity systems are inserted in a network testbed of air force research laboratory to identify attack sessions in normal network activities in real-time. In the year 2000, further experiments were conducted and three additional datasets were created, known as 2000 scenario-specific datasets. Both scenarios include a Distributed Denial of Service (DDoS) attack executed by an intruder. These scenarios contain various sessions of audit and network traffic. In these sessions, a DDoS software is installed by an attacker on a compromised host to execute a DDoS attack on a remote server. The only difference between these two scenarios is the difference in the attacker's role, i.e., the attacker in the second scenario is more stealthy than the attacker in the first scenario. Mixed reviews are found on how useful this dataset can be to train a cybersecurity system in [44]. For example, a team of researchers conducted experiments and found this dataset useful with Snort in Cisco intrusion detection systems in [111]. However, recommendations were also made to make this dataset more real so that it can support latest cybersecurity systems. On the other hand, there was another team of researchers who concluded that Snort could not perform well due to limited information of attacks in this dataset in [12]. This team suggested that the performance of a cybersecurity system can be improved if this dataset was combined with other datasets. Based on these mixed reviews, it can be concluded that this dataset needs improvements and information of latest cyber-attacks needs to be included in it.

4.1.2 ***ADFA Dataset.*** The Australian Defence Force Academy (ADFA) cybersecurity datasets contain data from Linux and Windows platforms [22]. These are designed to evaluate the performance of Host-based Intrusion Detection Systems (HIDS). The HIDS can monitor and analyze the internal environment of a computing system and incoming and outgoing network packets on its network interfaces. These datasets are freely available for academic research. These datasets are developed to better represent a compromised system from an initial behavior to a final compromised response and set a more realistic benchmark to evaluate the performance of a cybersecurity system. When designing these datasets, industry penetration testing methodologies were used and a new attack framework was created as shown in Fig. 2. This framework gives an understanding to readers and users of this dataset on how a sample attack framework works and how this dataset can be

used to deal with different types of cyber-attacks. This framework explains a connection between link vulnerabilities and their effects starting by focusing on deployability phases and effects of a particular attack. It provides a sample model to developers of cybersecurity systems to understand the scope of a cybersecurity dataset and its coverage on various types of attacks. Different layers of this framework, i.e., vulnerabilities, focus, accessibility, vector, deployability, and effect chain, explain the entire idea of how an attack can happen and how to deal with it to stop possible damages to an underlying network-based system. The main purpose of this sample framework is to modernize a cybersecurity system, plan a methodology based on current attack practices, use current industrial testing methodologies, and design datasets that can be used to train a cybersecurity system. Researchers in the cybersecurity community started using this dataset soon after its release. This dataset was used with a one-class SVM to detect intrusions in the cybersecurity system in [124]. However, it was found during experiments that this dataset was not robust against all types of cyber-attacks. Another attempt was made to extract and analyze specific features from this dataset to build an adaptive cybersecurity system in [123]. This extraction and analysis of features showed an acceptable performance, however, it was not suitable for modern complex cybersecurity systems. These references show that this dataset still needs further improvements to match the requirements of current cybersecurity systems.

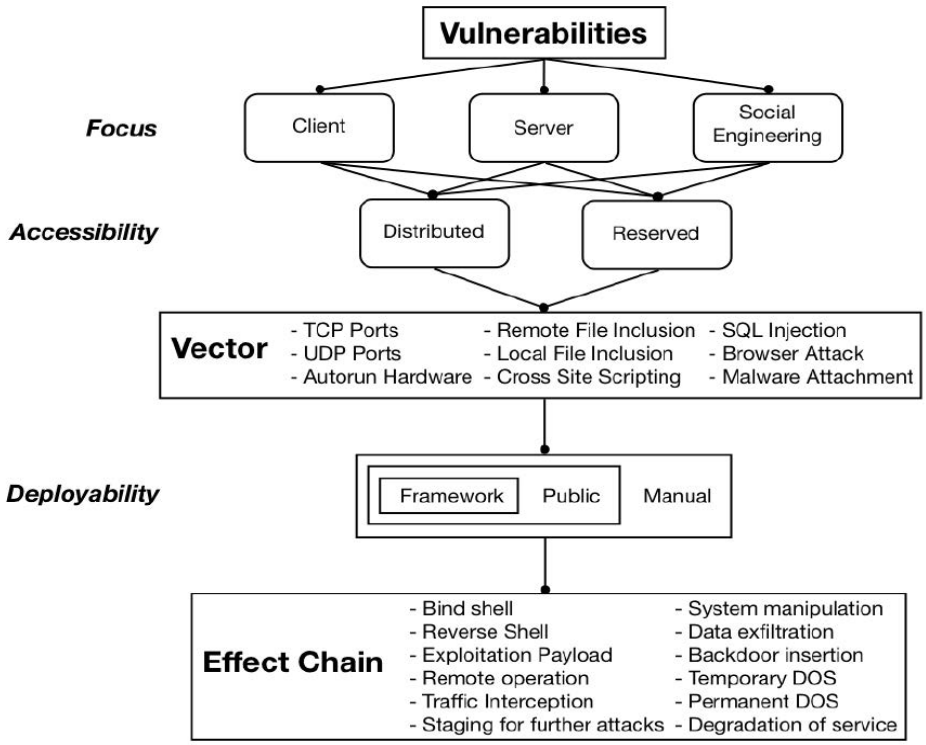

Fig. 2. Attack framework [21]

4.1.3 ***NetReSec pcap Files.*** The Network Forensics and Network Security Monitoring (NetReSec) is an independent software vendor that focuses on network security [1]. This vendor specializes

in software for network forensics and analysis. This company maintains a pool of pcap files and network traces that are freely available. This pool provides a useful resource to perform network evaluations on a cybersecurity system. The provided pcap files and network traces are classified into multiple categories, e.g., cyber defense exercises, malware traffics, network forensics, and packet injection attacks. Contents of each category are captured from different resources and can be used for various purposes. For example, the contents of cyber defense exercises include network traffics from different resources for cyber-exercises and competitions. The contents of malware traffic category are captured from honeypots [105], sandboxes [84] and real-world intrusions, and can be used to analyze web-based malware attacks. In the recent years, these pcap files were used with different machine learning algorithms, e.g., k-NN, SVM, and Bayesian networks, to train cybersecurity systems to detect different types of attacks [42, 106, 117]. The systems showed good performances in the presence of specific attacks. However, the main purpose was to analyze the performance of employed protocols, domain names and communication patterns in the presence of specific attacks in the underlying networks and a detailed analysis of modern attacks was missing.

4.1.4 ***Cyber Research Center Datasets.*** The Cyber Research Center (CRC) from the United States military academy offers datasets for public use in cybersecurity research [3]. These datasets provide a mean to match IP addresses from pcap files to IP addresses in internal networks. These datasets are distributed into four categories, i.e., Snort intrusion detection logs, Domain Name Service (DNS) logs, web server logs, and Splunk log server aggregate logs. The Snort are used to analyze real-time network traffic along with logging of data packets. It is an open-source system and its detection logs can be used to analyze network traffic to protect an underlying system from malware attacks. The Snort gained popularity due to its accurate detection of threats at high speeds and is considered as a suitable intrusion prevention technology worldwide. The DNS logs are distributed into two categories, i.e., external DNS service logs and message logs. The web server logs are also distributed into two categories, i.e., Apache web server access logs and Apache web server error logs. The Splunk is a security information and event management tool to analyze and aggregate security logs from different applications and solutions in the deployed environment. Later, the collected log records can be used to get an aggregated view and real-time monitoring of security events within the monitored environment. These datasets were used to detect malware and design efficient intrusion detection systems in [77, 83, 95, 102]. The designed intrusion detection systems were trained using CRC datasets and detected basic level of attacks, however, they do not stand against modern attacks due to limited available information in these datasets.

4.1.5 ***Creation and Extension of Datasets with Packet Capturing Tools.*** Packet Capture Data, formally known as pcap, is an application programming interface that captures network packets arriving at or transmitting from an Ethernet port. There are many packet capture libraries, e.g., Libpcap and WinPCap [24], that can be used by various network analyzing tools to analyze the behavior of and traffic generated by a network. Some very popular tools are WireShark [81], Nmap [27], tcpdump [33], and NetFlow [29]. The Internet engineering task force has listed one hundred and forty-four IP addresses. This list includes many popular protocols, e.g., Internet control message protocol [88], user datagram protocol [56], and transmission control protocol [10]. Applications and user programs use these popular protocols to generate network packets to transmit data over the Internet. Before transmission, an Ethernet frame is composed at the physical layer. This frame consists of a header and a payload. The header contains a medium access control address and the payload contains data that need to be transmitted. To transmit data over the Internet, the payload must also contain an IP header. Some data or other encapsulated protocols may be encapsulated in the IP payload. This entire Ethernet frame can be captured by pcap for further analysis and feature extraction. In the machine learning domain, algorithms depend on the training and testing datasets.

These datasets are required to test the performance of designed systems and algorithms. Although standard datasets are available to train machine learning algorithms in the cybersecurity domain, these datasets do not contain most updated data. Furthermore, it is not necessary that these datasets may match the traffic generated by all types of networks. Each network has a different size and deal with different types of nodes and applications. These applications generate different types of network traffic including multimedia and non-multimedia data. The packet capturing tools, e.g., pcap, WireShark, Nmap, tcpdump, and NetFlow, are needed to capture real-time network data generated by various applications running on a specific network. This captured data can be used to train and test representation learning algorithms used in cybersecurity systems.

4.2 Representation Learning for Cybersecurity

In this section, we discuss various techniques from recent literature. These techniques are based on representation learning algorithms and designed for different cybersecurity applications. These techniques are distributed into three broad categories, i.e., supervised, unsupervised, and deep architectures, as shown in Fig. 3. In each category, the techniques are discussed in detail with pros and cons in the following subsections.

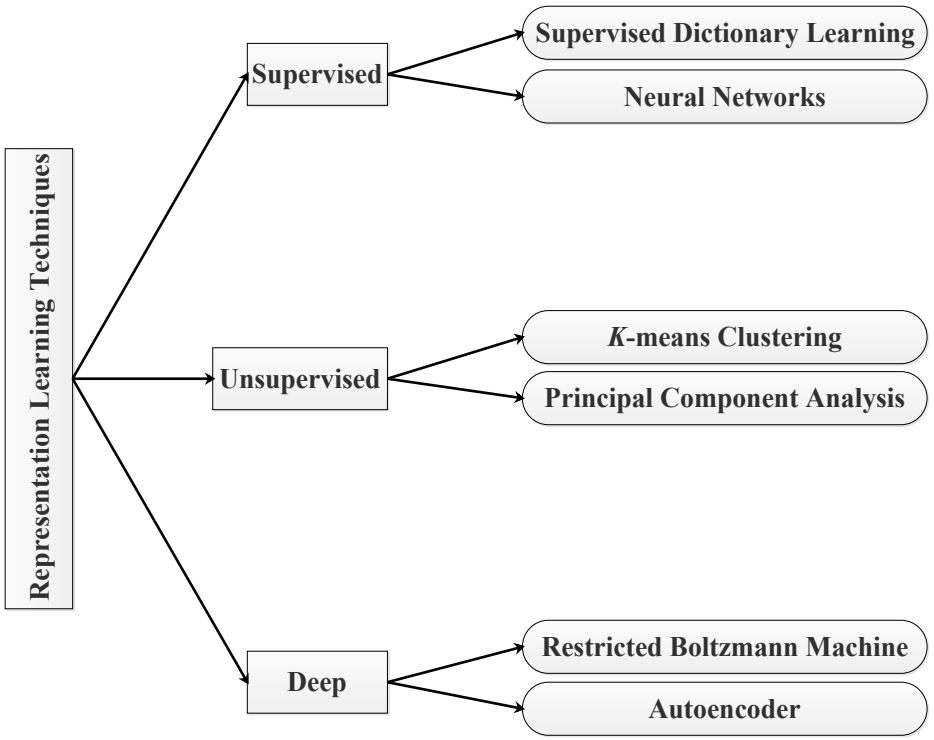

Fig. 3. Classification of techniques based on representation learning algorithms

4.2.1 *Supervised Representation Learning Algorithms.*

Supervised Dictionary Learning. In the recent years, supervised representation learning based techniques are used to investigate security-related issues in cybersecurity systems. A review on

recent advancements in machine learning techniques along with soft computing methodologies to secure Information and Communication Technology (ICT) based systems was presented in [15]. In this review, various security issues related to the Internet and its services for ICT systems are analyzed and trends of the supervised dictionary learning along with soft computing methodologies are investigated to secure ICT systems. Another survey on machine learning methods to protect cyber-manufacturing systems against cyber-attacks was presented in [122]. The cyber-manufacturing systems delineate the concept of combining various ICT systems, e.g., IoT, cloud computing, sensor networks, and machine learning platforms. However, these systems face various strong cyber-attacks including Stuxnet and computer numerical controlled milling attacks. In this survey, it is suggested through experiments that representation learning algorithms, especially supervised dictionary learning algorithms, can detect cyber-attacks with a high accuracy rate if physical data are studied properly. A survey on industrial cyber-physical systems along with their attack detection and security control mechanisms was presented in [25]. This survey discusses DoS, deception, and replay attacks along with their weaknesses. Furthermore, various developments on attack detection are reviewed from a detection perspective. These developments can be utilized to develop supervised dictionary learning algorithms to protect industrial cybersecurity systems.

Supervised dictionary learning models for intrusion detection were reviewed in [62]. In this work, feature selection techniques are suggested to construct better adversary-aware classifiers and a metric (i.e., model robustness score) is defined to evaluate relative resilience of different models. A novel methodology based on supervised dictionary learning to automatically identify integrity attacks was proposed for cybersecurity systems in [80]. In this method, a feature set is designed to sense properties of integrity attacks to train dictionary learning algorithms. Previously unseen attacks are also handled by adding a novelty detection component. A framework based on semi-supervised dictionary learning was introduced to identify Sybil nodes in [37]. In this framework, a small set of authorized and Sybil nodes is taken from a social network of nodes. Later, label information is propagated to remaining nodes to inform them about authorized and Sybil nodes.

A technique based on dual graph constraints was proposed to design a low-ranked dictionary learning algorithm for object classification systems in [32]. In this technique, low-dimensional space is used to train proposed dictionary learning algorithm to provide separability between intra and inter-classes. This technique shows better performance on small-sized datasets to classify objects and can be useful to detect malicious objects in cybersecurity systems. A similar dictionary learning approach based on projection property to identify human beings in captured videos was proposed in [128]. In this approach, feature projection metrics are combined with a set of dictionaries to classify labeled and unlabeled videos. Later, the labeled videos are used to support learned dictionaries. The main purpose of this approach is person re-identification along with detecting malicious activities in cybersecurity systems.

Artificial Neural Networks. Artificial Neural Networks (ANNs) are inspired from biological neural networks to perform different tasks [98]. The tasks are performed through learning based on provided examples. In the learning process, there are no specific rules. The ANNs consist of various small units called artificial neurons where each neuron can communicate with other connected neurons. The connections between neurons are called edges. The neurons and edges have specific weights that increase or decrease as the learning process proceeds. Generally, the neurons are distributed into multiple layers to perform different operations on input data. Nowadays, the ANN is a standard and popular example in different domains of data processing, e.g., computer vision, pattern recognition, speech processing, social networks, data classification, video games, object tracking, big data analysis, and cybersecurity.

There are many variants of ANNs, e.g., Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Deep Belief Networks (DBNs), etc. The CNNs consist of one or multiple convolutional layers where each layer consists of multiple fully connected layers [64]. This architecture is designed to process two-dimensional data and has many applications in the image and audio processing domains. In the RNNs, the connections between nodes are sequential graphs [38]. These networks are basically used to process sequential data and have applications in unsegmented data analysis and speech recognition. The RNNs have two broad classes of networks called finite and infinite impulses. The finite impulse class is based on a directed acyclic graph that can be unrolled. On the other hand, the infinite class possesses opposite features, i.e., it is based on a directed cyclic graph and cannot be unrolled. The DBNs consist of multiple layers of hidden units [66]. There are no connections between units within a layer, however, there are connections between layers. The DBNs can be used in both categories, i.e., supervised and unsupervised learning. The DBN is a popular choice in many applications, e.g., pattern recognition, drug discovery, and electroencephalography.

To secure networking, social computing, and cybersecurity systems, a risk assessment technique based on a back-propagation neural network was proposed in [70]. In this technique, an improved cuckoo search algorithm is used to train the network to improve the accuracy and stability in information security risk assessment processes in cybersecurity systems. A technique based on ANNs was proposed to detect known and unknown DDoS attacks in [19, 94]. In this technique, the DDoS attacks are detected based on specific patterns and the main purpose is to separate the traffic of DDoS attacks from genuine traffics in the cybersecurity systems. A multi-model based framework was proposed to analyze observed data related to cyber-attacks and make predictions about the progress of adversaries in [86]. This framework consists of multiple predictive models, i.e., a non-linear auto-regressive model, a non-linear auto-regressive exogenous model, an auto-regressive neural network, and an auto-regressive integrated moving average model, to make predictions about adversarial movements with a reliable accuracy.

A context information-based cybersecurity defense system was proposed to protect power systems in [100]. In this system, a vicious fault in an underlying local area network is identified by analyzing context information. Measurements are fed into a probabilistic neural network to predict the fault. An ensemble modeling based approach to detect integrity attacks in cybersecurity systems was proposed in [79]. This approach is a combination of an ANN and a linear time-invariant modeling and is tested against different integrity attacks with varying intensity levels.

A reservoir computing architecture based on neuromorphic computing was proposed to detect anomalies in cybersecurity systems in [69]. The architecture is proposed for feedback-based systems and its application is introduced in smart grids for anomaly detection. Another similar architecture to detect anomalies in automobiles was proposed in [110]. In this architecture, multivariate Markov chains are combined with RNNs to create anomaly detectors for automotive cyber-attacks.

4.2.2 ***Unsupervised Representation Learning Techniques.***

k-Means Clustering. *k*-means clustering is a technique to analyze and partition input data (i.e., observations) into *k*-clusters [59]. The *k*-means clustering is considered a computationally complex problem, however, efficient heuristic algorithms can be deployed for quick convergences. Formed clusters can have different shapes and sizes. This technique is different from *k*-nearest problem in the machine learning domain. An *n*-nearest neighbor algorithm can be applied on a cluster formed from a *k*-means clustering technique to classify newly entered data in the existing clusters. The *k*-means clustering can be applied in different applications including computer vision, astronomy, marketing and agriculture.

A novel unsupervised classification approach was proposed to detect network anomalies in [126]. In this approach, the k-means clustering is combined with iterative decision trees to classify normal and abnormal activities in network traffic. A machine learning technique based on a hierarchical k-means clustering was proposed to monitor traffic on the Internet to detect cyber-attacks in [78]. This technique analyzes partitioned clusters by using traffic analysis profile feature vectors to monitor typical malicious behaviors. A hybrid k-means clustering approach was proposed to deal with cyber-activities including reconnaissance and corporate cyber scanning activities in [11]. In this approach, datasets consist of real network traffic are used in two experimental environments to test the performance against unsupervised k-means clustering and expectation maximization approaches.

Principal Component Analysis. Principal Component Analysis (PCA) is a procedure to convert a set of observations to a set of linearly uncorrelated variables called principal components [57]. In converted principal components, the largest possible variance is held by the first component and each succeeding component holds the highest variance value. The PCA is the simplest form of eigen vector based analysis and is used as a tool to analyze data and make predictive models. The PCA is quite popular in neuroscience and quantitative finance fields.

Various surveys on cybersecurity systems, fourth and fifth generation cellular networks, and smart power grids, were presented to discuss authentication and privacy-preservation against different cyber-attacks in [30, 40, 120], respectively. These surveys highlight different frameworks and machine learning tools for data analysis and future research directions, e.g., how to design and model different types of attacks, design strategies for various attacks through risk assessments, and design testbeds to validate security issues and solutions.

An approach based on distributed SVMs was proposed to detect the injection of stealthy false data in smart grids in [28]. This approach uses PCA to minimize data dimensions that may cause computational complexities during data processing. A real-time algorithm was proposed to detect abnormal changes in power networks data stored in accessible databases in [115]. In this algorithm, the PCA is used to monitor power flow results and identify abnormal input data modified by cyber-attacks.

A distributed blind intrusion detection framework was proposed for cybersecurity systems in [93]. In this framework, a PCA-based approach is used to detect intrusions by analyzing sensor measurements and statistical properties of graph-signals. To detect anomalies in large-scale networks, a technique based on geometric area analysis was proposed in [75]. This technique is based on trapezoidal area estimation to observe computed parameters, and dimensions of network data are reduced by the PCA. An inference-based intrusion detection approach was proposed to identify cyber-attacks in software-defined networks in [4]. In this approach, regularly labeled flows generated by software-defined networks are analyzed to identify cyber-attacks and dimensions of data are reduced by the PCA.

4.2.3 ***Deep Representation Learning Techniques.***

Restricted Boltzmann Machine. Restricted Boltzmann Machine (RBM) is based on a generative stochastic ANN [49]. The RBM can work in both supervised and unsupervised modes. In the RBMs, nodes from each group of units may share a symmetric connection, however, there are no connections between nodes within the same group of units. The RBMs can be used to form deep learning networks by stacking multiple RBMs. The formed deep learning networks can later be tuned using gradient descent and back-propagation networks. The RBMs can be used in different practical applications, e.g., feature learning, data classification, and reducing dimensions of collected data.

There are many existing surveys in literature to highlight the use of deep learning techniques in various applications related to cybersecurity. A survey on various applications of deep learning was presented in [76]. In this survey, various deep learning techniques including RBMs are discussed to address some important problems in big data analytics and security in cybersecurity systems. Similarly, another survey on computational intelligence and analytics techniques for cybersecurity systems was presented in [55]. In this survey, various deep learning techniques including RBMs are discussed to analyze big data for computational intelligence along with recent advancements and applications.

Various recent developments using machine learning methods in software-defined networks were reviewed to implement intrusion detection systems for cybersecurity applications in [107]. This study focuses on deep learning techniques including RBMs and tools that can be used to design intrusion detection systems for software-defined networks. Various deep learning methodologies, e.g., RNNs, deep neural network, and RBMs-based DBNs, were reviewed in [65]. In this study, various deep learning methodologies along with other machine learning techniques are discussed from a network anomaly detection perspective. Experiments are also conducted to check the compatibility of deep learning methodologies to analyze network traffic.

A heterogeneous deep learning framework was proposed to intelligently detect malware in network traffic in [127]. This heterogeneous framework consists of an autoencoder, multi-layered RBMs, and associative memory layers, to identify unknown malware. A linear approach was proposed to increase the productivity of predictive manufacturing systems along with resilience and interoperability features in [68]. In this approach, features are extracted through an RBM. Performances of prediction processes are also improved through the same RBM by building an intelligent manufacturing system to automatically predict and reconfigure faulty events.

Autoencoder. An autoencoder is based on an ANN and operates in an unsupervised way to learn efficient data coding [116]. It is used to learn representations of input data for compression purposes. The compression is applied by reducing the dimensions of supplied data. The input data are transformed into a short code that is later uncompressed to match the original input data. The autoencoders can be used in a stacked form in certain applications, e.g., image recognition. In the stacked format, the lower layers learn and encode easy features while the upper layers analyze the output of previous layers and encode missing, difficult, or hidden features. This process continues until the entire data are encoded.

An improved extreme learning machine framework was proposed for smart grids to detect false data attacks in [125]. This framework uses an autoencoder to minimize the dimensions of measured data and is tested to effectively detect unobservable attacks. A deep learning based approach was proposed to detect distributed attacks in social IoT in [26]. In this approach, an autoencoder is used to discover hidden patterns from training data to distinguish attacks from benign traffic. This approach performs well against centralized attack detection approaches. A framework was proposed to detect impersonation attacks in Wi-Fi signals in IoT in [6]. In this framework, a stacked autoencoding is used to provide meaningful representations from a well-referenced Wi-Fi dataset, i.e., Aegean Wi-Fi intrusion dataset. A robust sensitivity-based learning algorithm was proposed to detect evasion attacks on classifiers in [16]. This algorithm is based on a modified stacked autoencoder and performs better against conventionally stacked and denoising autoencoders in terms of time complexity.

5 LIMITATIONS, RESEARCH CHALLENGES, AND FUTURE RESEARCH DIRECTIONS

In Section 4, we describe many efforts that are made using representation learning algorithms in the cybersecurity domain. It is clearly shown that cross-disciplinary approaches have become

popular in recent years. Along with algorithms, we also discuss various datasets that can be used by representation learning algorithms in the cybersecurity domain. In the following subsections, we highlight major limitations of presented efforts and datasets. Based on their limitations, we highlight various research challenges and future research directions.

5.1 Limitations of DataSets

Datasets explained in Section 4.1.1 were designed long time ago. Their major limitations are listed below.

- They are predefined datasets. They contain old data readings. These readings do not match the requirements of current cybersecurity applications.
- These datasets are quite big in size. Processing of these datasets using representation learning algorithms is not only a time-consuming process but requires computational and storage resources. Furthermore, processing of old data readings is a waste of time and computational and storage resources.
- These datasets are publicly available and anyone can access them. If they are used in a cybersecurity system, attackers can easily get access to the targeted system by analyzing the datasets being used.
- These datasets are very old and do not contain any information about current generation of cyber-attacks. As a result, a cybersecurity application based on these datsets cannot protect an underlying system from current cyber-attacks.

5.2 Research Challenges in DataSets

When using the datasets explained in Section 4.1.1 to train representation learning algorithms for cybersecurity applications, researchers and developers can face following research challenges.

- It is hard to find a dataset matching the requirements of underlying cybersecurity systems or applications.
- The size of a dataset is directly proportional to the rate of accuracy. However, processing a large size dataset is not only a time-consuming task but requires abundant computational and storage resources. As a result, a balance needs to be maintained between the size of a dataset and accuracy level for real-time cybersecurity applications.
- To provide a real-time support and save costs of computational and storage resources, large-sized datasets are usually distributed into small-sized datasets. However, this distribution affects the accuracy ratio. Useful features can easily be missed during this distribution, therefore, care must be taken when dividing a large-sized dataset into small datasets.
- Public datasets contain different types of network traces, log files, and operating system level data. To protect a particular system or application, the researchers and developers need to track and process specific data files.

5.3 Future Research Directions for Datasets

In order to use datasets for cybersecurity applications, following points can be considered as future research directions.

- Instead of relying on existing datasets that do not match the requirements of a cybersecurity system or application, the researchers and developers can design their own datasets. These datasets must contain system specific data. It will be easy to manage and update these datasets. Furthermore, access to these datasets will be restricted which not only provides protection but also can preserve the privacy of underlying systems.

- Applications in the cybersecurity domain keep changing on daily bases. Some of them might not contain very sensitive data. To support real-time processing, the applications can be divided into different classes based on their sensitivity and requirements. Highly sensitive and real-time applications need to be given high priorities and data generated by these applications must be kept in datasets to keep the cybersecurity system updated.
- For a cybersecurity system to perform effectively, it is very important to have an access to network and kernel-level data. The network data contain information about incoming data traffic while the kernel data contain information about system calls and network and security logs. An accurate estimation of an intrusion attack can be made by analyzing the kernel-level data. The data are always changing from time-to-time, therefore, the data need to be added and deleted frequently in datasets to keep the cybersecurity system updated and alerted.
- To test the accuracy of a representation learning algorithm, it is necessary to have a synchronization between training and testing datsets. The performance needs to be tested against the same datasets. On a theoretical level, this comparison is accepted. However, in practical, it may not be possible. Training and testing data change frequently in real-time applications, therefore, current network traces need to be added in training and testing datasets to test the accuracy of representation learning algorithms being used. Some algorithms might not be compatible with rapidly changing datasets, therefore, it is very important to select representation learning algorithms that can accommodate real-time changes.
- Training time plays a very important role in representation learning algorithms. Due to continuously changing nature and features, fresh updates about cyber-attacks need to be added in the training and testing datasets. Due to this addition, the cybersecurity systems need to be trained frequently. This continuous training requires powerful computational and storage resources, and as a result, extra costs and man power will be required to rent these resources and train cybersecurity systems, respectively.
- Classification timing is another critical factor to test the performance of a cybersecurity system based on a representation learning algorithm. Due to large-sized datasets, the classification may become time and resource consuming processes. The classified data help system administrators to analyze final outputs and take necessary actions. However, this analysis can only be performed after the classification is done. Long processing delays and late classification analysis may be suitable for offline systems, however, real-time systems can easily be attacked and compromised.

5.4 Limitations of Techniques

Representation learning based techniques summarized in Section 4.2 can be used in various daily life applications, e.g., healthcare, transportation management, cloud data management, person identification, etc. However, in cybersecurity applications, these techniques cannot be used directly and may face certain limitations. Some major limitations are listed below.

- In daily life applications, e.g., healthcare, transportation management, cloud data management, and person identification, the representation learning algorithms can be trained once in a while and used for a long time without a retraining. The training time is usually quite long. On the other hand, real-time cybersecurity applications require frequent training of these algorithms and the availability of updated datasets, and as a result, most of these algorithms might not be suitable for real-time cybersecurity applications.
- Frequent training might be required on a daily basis or more than once a week due to rapidly changing nature and features of cyber-attacks. Whenever a new attack is identified and its features and patterns become known, a retraining is required. The representation learning

algorithms usually train from scratch and require many days to finish the training process. Such a long training process wastes time and computational and storage resources if small changes are required to be made on a daily basis.

5.5 Research Challenges in Techniques

When using representation learning algorithms for cybersecurity applications, researchers and developers may face the following research challenges.

- Existing representation learning algorithms are quite complex and might not support real-time processing.
- Cost becomes another challenge when executing and managing existing representation learning algorithms.
- Existing algorithms are basically designed for specific applications, e.g., pattern recognition, computer vision, and image and signal processing. Using these algorithms for cybersecurity applications require significant changes.
- Due to the complexity of these algorithms and the availability of cheap resources, most of these algorithms are executing on public cloud platforms. Although cloud service providers ensure best effort services, still their platforms cannot be fully trusted. Granting access to applications running on public cloud platforms to access sensitive data can open doors for intruders. Furthermore, the data travel over public networks, i.e., the Internet, and as a result, the privacy can easily be compromised.

5.6 Future Research Directions for Representation Learning Techniques

In order to use representation learning algorithms for cybersecurity applications, the following points need to be considered as future research directions.

- Most organizations do not have sufficient budget to purchase expensive cybersecurity systems and applications. Furthermore, some organizations might be dealing with real-time data and applications. To propose cost-effective and real-time cybersecurity systems based on representation learning algorithms, researchers and developers need to use or develop representation learning algorithms that must require less time and computational resources to finish the training process.
- In certain research domains, e.g., healthcare, transportation management, cloud data management, and person identification, it is easy to obtain training and testing datasets. There are many datasets that are publicly available. However, in the cybersecurity domain, the availability of most recent training and testing data is not easy. Furthermore, the organizations do not prefer to share their network traffic to external users and researchers, as it may contain sensitive information and records. In this situation, there is a need to develop lightweight representation learning algorithms that can be trained locally.
- Due to the continuously changing nature of cyber-attacks, it is very important to have authentic and versatile data from different resources (e.g., different cybersecurity systems) for training and testing purposes. Global organizations discussed in Subsection 2.2 can offer their help and share resources. However, there is a need for joint collaborations between different organizations on a local level.
- To produce accurate results, the representation learning algorithms need to use huge volumes of data for training and testing purposes. Real-time cybersecurity applications generate huge volumes of data on daily bases. This data need to be labeled properly if supervised or semi-supervised algorithms are used. To increase the accuracy ratio, correct data labeling is the preliminary step. In the cybersecurity domain, chances of the availability of labeled

data are very low and labeling of large volumes of data requires lots of computational and human resources. Therefore, there is a need for representation learning algorithms that can automatically label huge volumes of data in real-time without requiring extra computational resources.

6 CONCLUSION

Representation learning is a sub-domain of machine learning that allows automatic discovery of features from raw input data to perform data classification and analysis. In machine learning tasks, data classification requires mathematically and computationally convenient input data. However, real-time multimedia and non-multimedia data, e.g., images, audio, videos, and sensor data, do not define specific features. The representation learning techniques offer an alternative by discovering features or representations through examination without relying on explicit algorithms. In this survey, we have discussed various cyber-attacks and initiatives taken by international organizations. In the cybersecurity domain, real-time applications deal with both multimedia and non-multimedia data. To process the data produced by various real-time applications, we have provided an in-depth overview of various representation learning computing platforms. These computing platforms are introduced by well-known vendors, e.g., IBM, Microsoft, Google, Amazon and Big ML. We have also discussed various datasets that can be utilized by representation learning algorithms in the cybersecurity domain. Later, we have discussed and summarized recent efforts made for cybersecurity systems by using representation learning algorithms. These efforts are classified into three broad categories, i.e., supervised, unsupervised, and deep architectures. In the end, we highlight various limitations in the available datasets and existing representation learning based techniques. The main purpose of highlighting the limitations is to tell the researchers and developers that there are still many open research challenges that need to be addressed when using the available datasets and techniques. These limitations also highlight various future research directions. These research directions highlight various facts that need to be considered to improve various features of available representation learning techniques to make them compatible with real-time applications.

REFERENCES

[1] 2018. Network Forensics and Network Security Monitoring (Netresec). http://www.netresec.com/?page=AboutNetresec

[2] Abebe Abeshu and Naveen Chilamkurti. 2018. Deep Learning: The Frontier for Distributed Attack Detection in Fog-to-Things Computing. *IEEE Communications Magazine* 56, 2 (2018), 169–175.

[3] United States Military Academy. 2018. Cyber Research Center. https://www.usma.edu/crc/SitePages/Home.aspx

[4] Ahmed AlEroud and Izzat Alsmadi. 2017. Identifying cyber-attacks on software defined networks: An inference-based intrusion detection approach. *Journal of Network and Computer Applications* 80 (2017), 152–164.

[5] Riham AlTawy and Amr M Youssef. 2016. Security tradeoffs in cyber physical systems: A case study survey on implantable medical devices. *IEEE Access* 4 (2016), 959–979.

[6] Muhamad Erza Aminanto, Rakyong Choi, Harry Chandra Tanuwidjaja, Paul D Yoo, and Kwangjo Kim. 2018. Deep Abstraction and Weighted Feature Selection for Wi-Fi Impersonation Detection. *IEEE Transactions on Information Forensics and Security* 13, 3 (2018), 621–636.

[7] Haiyong Bao, Rongxing Lu, Beibei Li, and Ruilong Deng. 2016. BLITHE: Behavior rule-based insider threat detection for smart grid. *IEEE Internet of Things Journal* 3, 2 (2016), 190–205.

[8] Cristóbal Barba-González, José García-Nieto, María del Mar Roldán-García, Ismael Navas-Delgado, Antonio J Nebro, and José F Aldana-Montes. 2019. BIGOWL: Knowledge centered Big Data analytics. *Expert Systems with Applications* 115 (2019), 543–556.

[9] Roger Barga, Valentine Fontama, Wee Hyong Tok, and Luis Cabrera-Cordon. 2015. *Predictive analytics with Microsoft Azure machine learning*. Springer.

[10] Miguel Barreiros and Peter Lundqvist. 2015. *QoS-Enabled networks: Tools and foundations*. John Wiley & Sons.

[11] Elias Bou-Harb, Mourad Debbabi, and Chadi Assi. 2013. A systematic approach for detecting and clustering distributed cyber scanning. *Computer Networks* 57, 18 (2013), 3826–3839.
[12] S Terry Brugger and Jedidiah Chow. 2007. An assessment of the DARPA IDS Evaluation Dataset using Snort. *UCDAVIS department of Computer Science* 1, 2007 (2007), 22.
[13] Anna L Buczak and Erhan Guven. 2016. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials* 18, 2 (2016), 1153–1176.
[14] Mike Burmester, Emmanouil Magkos, and Vassilis Chrissikopoulos. 2012. Modeling security in cyber–physical systems. *International journal of critical infrastructure protection* 5, 3-4 (2012), 118–126.
[15] Francesco Camastra, Angelo Ciaramella, and Antonino Staiano. 2013. Machine learning and soft computing for ICT security: an overview of current trends. *Journal of Ambient Intelligence and Humanized Computing* 4, 2 (2013), 235–247.
[16] Patrick PK Chan, Zhe Lin, Xian Hu, Eric CC Tsang, and Daniel S Yeung. 2017. Sensitivity based robust learning for stacked autoencoder against evasion attack. *Neurocomputing* 267 (2017), 572–580.
[17] Brijesh Kashyap Chejerla and Sanjay K Madria. 2017. QoS guaranteeing robust scheduling in attack resilient cloud integrated cyber physical system. *Future Generation Computer Systems* 75 (2017), 145–157.
[18] Ping Chen, Lieven Desmet, and Christophe Huygens. 2014. A study on advanced persistent threats. In *IFIP International Conference on Communications and Multimedia Security*. Springer, 63–72.
[19] Sujit Rokka Chhetri, Arquimedes Canedo, and Mohammad Abdullah Al Faruque. 2016. Kcad: kinetic cyber-attack detection method for cyber-physical additive manufacturing systems. In *Proceedings of the 35th International Conference on Computer-Aided Design*. ACM, 74.
[20] Chris Clifton and Tamir Tassa. 2013. On syntactic anonymity and differential privacy. In *Data Engineering Workshops (ICDEW), 2013 IEEE 29th International Conference on*. IEEE, 88–93.
[21] Gideon Creech. 2014. *Developing a high-accuracy cross platform Host-Based Intrusion Detection System capable of reliably detecting zero-day attacks.* Ph.D. Dissertation. University of New South Wales, Canberra, Australia.
[22] Gideon Creech and Jiankun Hu. 2013. Generation of a new IDS test dataset: Time to retire the KDD collection. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*. IEEE, 4487–4492.
[23] Kristopher Daley, Ryan Larson, and Jerald Dawkins. 2002. A structural framework for modeling multi-stage network attacks. In *Parallel Processing Workshops, 2002. Proceedings. International Conference on*. IEEE, 5–10.
[24] Luca Deri et al. 2004. Improving passive packet capture: Beyond device polling. In *Proceedings of SANE*, Vol. 2004. Amsterdam, Netherlands, 85–93.
[25] Derui Ding, Qing-Long Han, Yang Xiang, Xiaohua Ge, and Xian-Ming Zhang. 2018. A survey on security control and attack detection for industrial cyber-physical systems. *Neurocomputing* 275 (2018), 1674–1683.
[26] Abebe Abeshu Diro and Naveen Chilamkurti. 2017. Distributed attack detection scheme using deep learning approach for Internet of Things. *Future Generation Computer Systems* (2017).
[27] Zakir Durumeric, Eric Wustrow, and J Alex Halderman. 2013. ZMap: Fast Internet-wide Scanning and Its Security Applications.. In *USENIX Security Symposium*, Vol. 8. 47–53.
[28] Mohammad Esmalifalak, Lanchao Liu, Nam Nguyen, Rong Zheng, and Zhu Han. 2014. Detecting stealthy false data injection using machine learning in smart grid. *IEEE Systems Journal* (2014).
[29] Cristian Estan, Ken Keys, David Moore, and George Varghese. 2004. Building a better NetFlow. In *ACM SIGCOMM Computer Communication Review*, Vol. 34. ACM, 245–256.
[30] Mohamed Amine Ferrag, Leandros Maglaras, Antonios Argyriou, Dimitrios Kosmanos, and Helge Janicke. 2017. Security for 4G and 5G cellular networks: A survey of existing authentication and privacy-preserving schemes. *Journal of Network and Computer Applications* (2017).
[31] Prahlad Fogla, Monirul I Sharif, Roberto Perdisci, Oleg M Kolesnikov, and Wenke Lee. 2006. Polymorphic Blending Attacks.. In *USENIX Security Symposium*. 241–256.
[32] Homa Foroughi, Nilanjan Ray, and Hong Zhang. 2018. Object classification with joint projection and low-rank dictionary learning. *IEEE Transactions on Image Processing* 27, 2 (2018), 806–821.
[33] Felix Fuentes and Dulal C Kar. 2005. Ethereal vs. Tcpdump: a comparative study on packet sniffing tools for educational purpose. *Journal of Computing Sciences in Colleges* 20, 4 (2005), 169–176.
[34] Robin Gandhi, Anup Sharma, William Mahoney, William Sousan, Qiuming Zhu, and Phillip Laplante. 2011. Dimensions of cyber-attacks: Cultural, social, economic, and political. *IEEE Technology and Society Magazine* 30, 1 (2011), 28–38.
[35] Gigaom. 2013. This is interesting: A fraud-detection company built on Google's Prediction API. https://gigaom.com/2013/07/31/this-is-interesting-a-fraud-detection-company-built-on-googles-prediction-api/
[36] Jairo Giraldo, Esha Sarkar, Alvaro A Cardenas, Michail Maniatakos, and Murat Kantarcioglu. 2017. Security and privacy in cyber-physical systems: A survey of surveys. *IEEE Design & Test* 34, 4 (2017), 7–17.
[37] Neil Zhenqiang Gong, Mario Frank, and Prateek Mittal. 2014. Sybilbelief: A semi-supervised learning approach for structure-based sybil detection. *IEEE Transactions on Information Forensics and Security* 9, 6 (2014), 976–987.

[38] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*. IEEE, 6645–6649.
[39] Tim Greene. 2015. Biggest data breaches of 2015. *Network* 10 (2015), 14.
[40] BB Gupta and Tafseer Akhtar. 2017. A survey on smart power grid: frameworks, tools, security issues, and solutions. *Annals of Telecommunications* 72, 9-10 (2017), 517–549.
[41] Didem Gürdür and Fredrik Asplund. 2017. A Systematic Review to Merge Discourses: Interoperability, Integration and Cyber-Physical Systems. *Journal of Industrial Information Integration* (2017).
[42] Fariba Haddadi, Duc Le Cong, Laura Porter, and A Nur Zincir-Heywood. 2015. On the effectiveness of different botnet detection approaches. In *Information Security Practice and Experience*. Springer, 121–135.
[43] Adam Hahn, Aditya Ashok, Siddharth Sridhar, and Manimaran Govindarasu. 2013. Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid. *IEEE Transactions on Smart Grid* 4, 2 (2013), 847–855.
[44] Tarfa Hamed, Jason B Ernst, and Stefan C Kremer. 2018. A Survey and Taxonomy on Data and Pre-processing Techniques of Intrusion Detection Systems. In *Computer and Network Security Essentials*. Springer, 113–134.
[45] Simon Hansman and Ray Hunt. 2005. A taxonomy of network and computer attacks. *Computers & Security* 24, 1 (2005), 31–43.
[46] Haibo He and Jun Yan. 2016. Cyber-physical attacks and defences in the smart grid: a survey. *IET Cyber-Physical Systems: Theory & Applications* 1, 1 (2016), 13–27.
[47] Ryan Heartfield and George Loukas. 2016. A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks. *ACM Computing Surveys (CSUR)* 48, 3 (2016), 37.
[48] Ryan Heartfield, George Loukas, and Diane Gan. 2016. You are probably not the weakest link: Towards practical prediction of susceptibility to semantic social engineering attacks. *IEEE Access* 4 (2016), 6910–6928.
[49] Geoffrey E Hinton. 2012. A practical guide to training restricted Boltzmann machines. In *Neural networks: Tricks of the trade*. Springer, 599–619.
[50] Geoffrey E Hinton and Terrence Joseph Sejnowski. 1999. *Unsupervised learning: foundations of neural computation*. MIT press.
[51] Thorsten Holz, Moritz Steiner, Frederic Dahl, Ernst Biersack, Felix C Freiling, et al. 2008. Measurements and Mitigation of Peer-to-Peer-based Botnets: A Case Study on Storm Worm. *LEET* 8, 1 (2008), 1–9.
[52] Fei Hu, Yu Lu, Athanasios V Vasilakos, Qi Hao, Rui Ma, Yogendra Patil, Ting Zhang, Jiang Lu, Xin Li, and Neal N Xiong. 2016. Robust cyber–physical systems: concept, models, and implementation. *Future Generation Computer Systems* 56 (2016), 449–475.
[53] Vincent C Hu, D Richard Kuhn, David F Ferraiolo, and Jeffrey Voas. 2015. Attribute-based access control. *Computer* 48, 2 (2015), 85–88.
[54] Abdulmalik Humayed, Jingqiang Lin, Fengjun Li, and Bo Luo. 2017. Cyber-physical systems security—A survey. *IEEE Internet of Things Journal* 4, 6 (2017), 1802–1831.
[55] Rahat Iqbal, Faiyaz Doctor, Brian More, Shahid Mahmud, and Usman Yousuf. 2017. Big Data analytics and Computational Intelligence for Cyber–Physical Systems: Recent trends and state of the art applications. *Future Generation Computer Systems* (2017).
[56] Stuart Jacobs. 2011. *Engineering information security: the application of systems engineering concepts to achieve information assurance*. Vol. 14. John Wiley & Sons.
[57] Ian Jolliffe. 2011. Principal component analysis. In *International encyclopedia of statistical science*. Springer, 1094–1096.
[58] Klaus Julisch. 2013. Understanding and overcoming cyber security anti-patterns. *Computer Networks* 57, 10 (2013), 2206–2211.
[59] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. 2002. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 7 (2002), 881–892.
[60] Da-Yu Kao and Shou-Ching Hsiao. 2018. The dynamic analysis of WannaCry ransomware. In *Advanced Communication Technology (ICACT), 2018 20th International Conference on*. IEEE, 159–166.
[61] Stamatis Karnouskos. 2011. Stuxnet worm impact on industrial cyber-physical system security. In *IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society*. IEEE, 4490–4494.
[62] Ziv Katzir and Yuval Elovici. 2018. Quantifying the resilience of machine learning classifiers used for cyber security. *Expert Systems with Applications* 92 (2018), 419–429.
[63] Ratinder Kaur and Maninder Singh. 2014. A survey on zero-day polymorphic worm detection techniques. *IEEE Communications Surveys & Tutorials* 16, 3 (2014), 1520–1549.
[64] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.
[65] Donghwoon Kwon, Hyunjoo Kim, Jinoh Kim, Sang C Suh, Ikkyun Kim, and Kuinam J Kim. 2017. A survey of deep learning-based network anomaly detection. *Cluster Computing* (2017), 1–13.

[66] Nicolas Le Roux and Yoshua Bengio. 2008. Representational power of restricted Boltzmann machines and deep belief networks. *Neural computation* 20, 6 (2008), 1631–1649.
[67] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444.
[68] Jay Lee, Chao Jin, and Behrad Bagheri. 2017. Cyber physical systems for predictive production systems. *Production Engineering* 11, 2 (2017), 155–165.
[69] Jialing Li, Lingjia Liu, Chenyuan Zhao, Kian Hamedani, Rachad Atat, and Yang Yi. 2017. Enabling Sustainable Cyber Physical Security Systems Through Neuromorphic Computing. *IEEE Transactions on Sustainable Computing* (2017).
[70] Senyu Li, Fangming Bi, Wei Chen, Xuzhi Miao, Jin Liu, and Chaogang Tang. 2018. An Improved Information Security Risk Assessments Method for Cyber-Physical-Social Computing and Networking. *IEEE Access* (2018).
[71] Kevin Liao, Ziming Zhao, Adam Doupé, and Gail-Joon Ahn. 2016. Behind closed doors: measurement and analysis of CryptoLocker ransoms in Bitcoin. In *Electronic Crime Research (eCrime), 2016 APWG Symposium on*. IEEE, 1–13.
[72] WIRED Magazine. 2011. FORD, Google team up to make smarter cars. https://www.wired.com/2011/05/ford-google-prediction-api/
[73] Gary Miliefsky. 2008. Proactive network security system to protect against hackers. US Patent 7,346,922.
[74] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2012. *Foundations of machine learning*. MIT press.
[75] Nour Moustafa, Jill Slay, and Gideon Creech. 2017. Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks. *IEEE Transactions on Big Data* (2017).
[76] Maryam M Najafabadi, Flavio Villanustre, Taghi M Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. 2015. Deep learning applications and challenges in big data analytics. *Journal of Big Data* 2, 1 (2015), 1.
[77] E Allison Newcomb, Robert J Hammell, and Steve Hutchinson. 2016. Effective prioritization of network intrusion alerts to enhance situational awareness. In *Intelligence and Security Informatics (ISI), 2016 IEEE Conference on*. IEEE, 73–78.
[78] Hironori Nishikaze, Seiichi Ozawa, Jun Kitazono, Tao Ban, Junji Nakazato, and Jumpei Shimamura. 2015. Large-Scale Monitoring for Cyber Attacks by Using Cluster Information on Darknet Traffic Features. *Procedia Computer Science* 53 (2015), 175–182.
[79] Stavros Ntalampiras. 2015. Detection of integrity attacks in cyber-physical critical infrastructures using ensemble modeling. *IEEE Transactions on Industrial Informatics* 11, 1 (2015), 104–111.
[80] Stavros Ntalampiras. 2016. Automatic identification of integrity attacks in cyber-physical systems. *Expert Systems with Applications* 58 (2016), 164–173.
[81] Angela Orebaugh, Gilbert Ramirez, and Jay Beale. 2006. *Wireshark & Ethereal network protocol analyzer toolkit*. Elsevier.
[82] Hamed Orojloo and Mohammad Abdollahi Azgomi. 2017. A method for evaluating the consequence propagation of security attacks in cyber–physical systems. *Future Generation Computer Systems* 67 (2017), 57–71.
[83] Ramkumar Paranthaman and Bhavani Thuraisingham. 2017. Malware collection and analysis. In *Information Reuse and Integration (IRI), 2017 IEEE International Conference on*. IEEE, 26–31.
[84] Sebastien Pouliot. 2010. System and method for using sandboxes in a managed shell. US Patent 7,725,922.
[85] Abhilasha Singh Rathor, Amit Agarwal, and Preeti Dimri. 2018. Comparative Study of Machine Learning Approaches for Amazon Reviews. *Procedia Computer Science* 132 (2018), 1552–1561.
[86] Aunshul Rege, Zoran Obradovic, Nima Asadi, and Edward Parker. 2018. Predicting Adversarial Cyber Intrusion Stages Using Autoregressive Neural Networks. *IEEE Intelligent Systems* (2018).
[87] Zahoor-Ur Rehman, Sidra Nasim Khan, Khan Muhammad, Jong Weon Lee, Zhihan Lv, Sung Wook Baik, Peer Azmat Shah, Khalid Awan, and Irfan Mehmood. 2017. Machine learning-assisted signature and heuristic-based detection of malwares in Android devices. *Computers & Electrical Engineering* (2017).
[88] Rami Rosen. 2014. Internet control message protocol (ICMP). In *Linux Kernel Networking*. Springer, 37–61.
[89] Jitendra Kumar Rout, Anmol Dalmia, Kim-Kwang Raymond Choo, Sambit Bakshi, and Sanjay Kumar Jena. 2017. Revisiting Semi-Supervised Learning for Online Deceptive Review Detection. *IEEE Access* 5, 1 (2017), 1319–1327.
[90] Robin Ruefle, Audrey Dorofee, David Mundie, Allen D Householder, Michael Murray, and Samuel J Perl. 2014. Computer security incident response team development and evolution. *IEEE Security & Privacy* 12, 5 (2014), 16–26.
[91] Jukka Ruohonen, Sami Hyrynsalmi, and Ville Leppänen. 2016. An outlook on the institutional evolution of the European Union cyber security apparatus. *Government Information Quarterly* 33, 4 (2016), 746–756.
[92] Nasser R Sabar, Xun Yi, and Andy Song. 2018. A Bi-objective Hyper-Heuristic Support Vector Machines for Big Data Cyber-Security. *IEEE Access* 6 (2018).
[93] Hamidreza Sadreazami, Arash Mohammadi, Amir Asif, and Konstantinos N Plataniotis. 2017. Distributed Graph-based Statistical Approach for Intrusion Detection in Cyber-Physical Systems. *IEEE Transactions on Signal and Information Processing over Networks* (2017).
[94] Alan Saied, Richard E Overill, and Tomasz Radzik. 2016. Detection of known and unknown DDoS attacks using Artificial Neural Networks. *Neurocomputing* 172 (2016), 385–393.

[95] Benjamin Sangster, TJ O'Connor, Thomas Cook, Robert Fanelli, Erik Dean, Christopher Morrell, and Gregory J Conti. 2009. Toward Instrumenting Network Warfare Competitions to Generate Labeled Datasets.. In *CSET*.
[96] Andrea Saracino, Daniele Sgandurra, Gianluca Dini, and Fabio Martinelli. 2016. Madam: Effective and efficient behavior-based android malware detection and prevention. *IEEE Transactions on Dependable and Secure Computing* (2016).
[97] Nolen Scaife, Henry Carter, Patrick Traynor, and Kevin RB Butler. 2016. Cryptolock (and drop it): stopping ransomware attacks on user data. In *Distributed Computing Systems (ICDCS), 2016 IEEE 36th International Conference on*. IEEE, 303–312.
[98] Robert J Schalkoff. 1997. *Artificial neural networks*. Vol. 1. McGraw-Hill New York.
[99] Mischa Schmidt and Christer Åhlund. 2018. Smart buildings as Cyber-Physical Systems: Data-driven predictive control strategies for energy efficiency. *Renewable and Sustainable Energy Reviews* 90 (2018), 742–756.
[100] Su Sheng, WL Chan, KK Li, Duan Xianzhong, and Zeng Xiangjun. 2007. Context information-based cyber security defense of protection system. *IEEE Transactions on Power Delivery* 22, 3 (2007), 1477–1481.
[101] Peter W Singer and Allan Friedman. 2014. *Cybersecurity: What everyone needs to know*. Oxford University Press.
[102] Sidney C Smith, Robert J Hammell, Kin W Wong, and Carlos J Mateo. 2016. An Experimental Exploration of the Impact of Host-Level Packet Loss on Network Intrusion Detection. In *Cybersecurity Symposium (CYBERSEC), 2016*. IEEE, 13–19.
[103] Houbing Song, Glenn A Fink, and Sabina Jeschke. 2017. *Security and Privacy in Cyber-physical Systems: Foundations, Principles, and Applications*. John Wiley & Sons.
[104] Aditya K Sood and Richard J Enbody. 2013. Targeted cyberattacks: a superset of advanced persistent threats. *IEEE security & privacy* 11, 1 (2013), 54–61.
[105] Lance Spitzner. 2003. *Honeypots: tracking hackers*. Vol. 1. Addison-Wesley Reading.
[106] V Srihari and R Anitha. 2014. DDoS detection system using wavelet features and semi-supervised learning. In *International Symposium on Security in Computing and Communication*. Springer, 291–303.
[107] Nasrin Sultana, Naveen Chilamkurti, Wei Peng, and Rabei Alhadad. 2018. Survey on SDN based network intrusion detection system using machine learning approaches. *Peer-to-Peer Networking and Applications* (2018), 1–9.
[108] Paweł Szynkiewicz and Adam Kozakiewicz. 2017. Design and evaluation of a system for network threat signatures generation. *Journal of Computational Science* 22 (2017), 187–197.
[109] Wei Tan, Yushun Fan, Ahmed Ghoneim, M Anwar Hossain, and Schahram Dustdar. 2016. From the service-oriented architecture to the web API economy. *IEEE Internet Computing* 20, 4 (2016), 64–68.
[110] Adrian Taylor, Sylvain Leblanc, and Nathalie Japkowicz. 2018. Probing the limits of anomaly detectors for automobiles with a cyber attack framework. *IEEE Intelligent Systems* (2018).
[111] Ciza Thomas, Vishwas Sharma, and N Balakrishnan. 2008. Usefulness of DARPA dataset for intrusion detection system evaluation. In *Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2008*, Vol. 6973. International Society for Optics and Photonics, 69730G.
[112] Olivier Thonnard, Leyla Bilge, Gavin O'Gorman, Seán Kiernan, and Martin Lee. 2012. Industrial espionage and targeted attacks: Understanding the characteristics of an escalating threat. In *International workshop on recent advances in intrusion detection*. Springer, 64–85.
[113] Gina C Tjhai, Maria Papadaki, Steven M Furnell, and Nathan L Clarke. 2008. The problem of false alarms: Evaluation with snort and DARPA 1999 dataset. In *International Conference on Trust, Privacy and Security in Digital Business*. Springer, 139–150.
[114] Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. 2016. Stealing Machine Learning Models via Prediction APIs.. In *USENIX Security Symposium*. 601–618.
[115] Jorge Valenzuela, Jianhui Wang, and Nancy Bissinger. 2013. Real-time intrusion detection in power system operations. *IEEE Transactions on Power Systems* 28, 2 (2013), 1052–1062.
[116] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research* 11, Dec (2010), 3371–3408.
[117] Daniel Walnycky, Ibrahim Baggili, Andrew Marrington, Jason Moore, and Frank Breitinger. 2015. Network and device forensic analysis of android social-messaging applications. *Digital Investigation* 14 (2015), S77–S84.
[118] Marilyn Wolf and Dimitrios Serpanos. 2018. Safety and Security in Cyber-Physical Systems and Internet-of-Things Systems. *Proc. IEEE* 106, 1 (2018), 9–20.
[119] Fang-Jing Wu, Yu-Fen Kao, and Yu-Chee Tseng. 2011. From wireless sensor networks towards cyber physical systems. *Pervasive and Mobile computing* 7, 4 (2011), 397–413.
[120] Guangyu Wu, Jian Sun, and Jie Chen. 2016. A survey on the security of cyber-physical systems. *Control Theory and Technology* 14, 1 (2016), 2–10.

[121] Longfei Wu, Xiaojiang Du, and Jie Wu. 2016. Effective defense schemes for phishing attacks on mobile computing platforms. *IEEE Transactions on Vehicular Technology* 65, 8 (2016), 6678–6691.

[122] Mingtao Wu, Zhengyi Song, and Young B Moon. 2017. Detecting cyber-physical attacks in CyberManufacturing systems with machine learning methods. *Journal of Intelligent Manufacturing* (2017), 1–13.

[123] Miao Xie and Jiankun Hu. 2013. Evaluating host-based anomaly detection systems: A preliminary analysis of adfa-ld. In *Image and Signal Processing (CISP), 2013 6th International Congress on*, Vol. 3. IEEE, 1711–1716.

[124] Miao Xie, Jiankun Hu, and Jill Slay. 2014. Evaluating host-based anomaly detection systems: Application of the one-class svm algorithm to adfa-ld. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2014 11th International Conference on*. IEEE, 978–982.

[125] Liqun Yang, Yuancheng Li, and Zhoujun Li. 2017. Improved-ELM method for detecting false data attack in smart grid. *International Journal of Electrical Power & Energy Systems* 91 (2017), 183–191.

[126] Yasser Yasami and Saadat Pour Mozaffari. 2010. A novel unsupervised classification approach for network anomaly detection by k-Means clustering and ID3 decision tree learning methods. *The Journal of Supercomputing* 53, 1 (2010), 231–245.

[127] Yanfang Ye, Lingwei Chen, Shifu Hou, William Hardy, and Xin Li. 2017. DeepAM: a heterogeneous deep learning framework for intelligent malware detection. *Knowledge and Information Systems* (2017), 1–21.

[128] Xiaoke Zhu, Xiao-Yuan Jing, Liang Yang, Xinge You, Dan Chen, Guangwei Gao, and Yunhong Wang. 2017. Semi-supervised Cross-view Projection-based Dictionary Learning for Video-based Person Re-identification. *IEEE Transactions on Circuits and Systems for Video Technology* (2017).