Open access • Posted Content • DOI:10.20944/PREPRINTS202102.0340.V1

Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective — Source link

Iqbal H. Sarker, Iqbal H. Sarker

Institutions: Swinburne University of Technology, Chittagong University of Engineering & Technology

Published on: 16 Feb 2021

Topics: Deep belief network, Deep learning, Recurrent neural network, Artificial neural network and Restricted Boltzmann machine

SURVEY ARTICLE

Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective

Iqbal H. Sarker[1,2]

Received: 19 November 2020 / Accepted: 19 February 2021 / Published online: 20 March 2021

Abstract

Deep learning, which is originated from an artificial neural network (ANN), is one of the major technologies of today's smart cybersecurity systems or policies to function in an intelligent manner. Popular *deep learning* techniques, such as multi-layer perceptron, convolutional neural network, recurrent neural network or long short-term memory, self-organizing map, autoencoder, restricted Boltzmann machine, deep belief networks, generative adversarial network, deep transfer learning, as well as deep reinforcement learning, or their ensembles and hybrid approaches can be used to intelligently tackle the diverse cybersecurity issues. In this paper, we aim to present a *comprehensive overview* from the perspective of these neural networks and deep learning techniques according to today's diverse needs. We also discuss the *applicability* of these techniques in various *cybersecurity tasks* such as intrusion detection, identification of malware or botnets, phishing, predicting cyberattacks, e.g. denial of service, fraud detection or cyberanomalies, etc. Finally, we highlight several *research issues and future directions* within the scope of our study in the field. Overall, the ultimate goal of this paper is to serve as a reference point and guidelines for the academia and professionals in the cyber industries, especially from the deep learning point of view.

Keywords Cybersecurity · Deep learning · Artificial neural network · Artificial intelligence · Cyberattacks · Cybersecurity analytics · Cyber threat intelligence

Introduction

Due to the increasing popularity of internet-of-things (IoT) [1], and today's dependency on digitalization, various security incidents or attacks have grown rapidly in recent years. Malicious activities, malware or ransomware attack [2], zero-day attack [3], cryptographic attack, unauthorized access [4], denial of service (DoS) [4], data breaches [5], phishing or social engineering [6], or various attacks on IoT devices etc. are common nowadays. These types of security incidents or cybercrime can affect organizations and individuals, cause disruptions, as well as devastating financial losses. For example, a data breach costs 8.19 million USD for the United States [7] according to the IBM report, and the total annual cost of cybercrime to the global economy is 400 billion USD [8]. Cybercrimes are growing at an exponential rate that brings an alarming message for the cybersecurity professionals and researchers [9]. Thus, the security management tools having the capability of detecting and preventing such incidents in a *timely and intelligent way* is urgently needed, on which the overall national security of the business, government, and individual citizens of a country depends.

Typically, cybersecurity is characterized as a collection of technologies and processes designed to protect computers, networks, programs, and data against malicious activities, attacks, harm, or unauthorized access [10]. According to today's numerous needs, conventional well-known security solutions such as antivirus, firewalls, user authentication, encryption etc. may not be effective [11–14]. The key issue with these systems is that they are normally operated by a few security analysts, where data management is carried out in an ad hoc manner and can, therefore, not work

This article is part of the topical collection "Deep learning approaches for data analysis: A practical perspective" guest edited by D. Jude Hemanth, Lipo Wang and Anastasia Angelopoulou.

✉ Iqbal H. Sarker
msarker@swin.edu.au

[1] Swinburne University of Technology, Melbourne, VIC 3122, Australia

[2] Department of Computer Science and Engineering, Chittagong University of Engineering & Technology, Chittagong 4349, Bangladesh

intelligently according to the needs [15, 16]. On the other hand, in the sense of computing that seeks to operate in an intelligent manner for cybersecurity management, *data-driven learning techniques, e.g., deep learning,* have evolved rapidly in recent years, in which we are interested.

Deep learning (DL) is considered as a part of machine learning (ML) as well as artificial intelligence (AI), which is originated from an artificial neural network (ANN) and one of the major technologies of the Fourth Industrial Revolution (Industry 4.0) [9] [17]. The worldwide popularity of “Cyber security” and “Deep learning” is increasing day-by-day, which is shown in Fig. 1. The popularity trend in Fig. 1 is shown based on the data collected from Google Trends over the last 5 years [18]. In this paper, we take into account ten popular neural network and deep learning techniques including supervised, semi-supervised, unsupervised, and reinforcement learning in the context of cybersecurity. These are (i) multi-layer perceptron (MLP), (ii) convolutional neural network (CNN or ConvNet), (iii) recurrent neural network (RNN) or long short-term memory (LSTM), (iv) self-organizing map (SOM), (v) auto-encoder (AE), (vi) restricted Boltzmann machine (RBM), (vii) deep belief networks (DBN), (viii) generative adversarial network (GAN), (ix) deep transfer learning (DTL or deep TL), and (x) deep reinforcement learning (DRL or deep RL). These deep neural network learning techniques or their ensembles and hybrid approaches can be used to intelligently solve different cybersecurity issues, such as intrusion detection, identification of malware or botnets, phishing, predicting cyber-attacks, e.g. DoS, fraud detection, or cyber-anomalies. Deep learning has its benefits to build the security models due to its better accuracy, especially learning from large quantities of security datasets [19]. The contribution of this paper is summarized as follows:

- This study concentrates on the knowledge of ANN and DL techniques, a part of artificial intelligence (AI), to function in a timely, automated, and intelligent manner in the context of cybersecurity, which are considered as the major technologies of the Fourth Industrial Revolution (Industry 4.0).
- We discuss various popular neural network and deep learning techniques including supervised, unsupervised, and reinforcement learning in the context of cybersecurity, as well as the applicability of these techniques in various cybersecurity tasks.
- Finally, we highlight several research issues and future directions within the scope of our study for future development and research in the domain of cybersecurity.

This paper is organized as follows. Section 2 provides a brief overview of cybersecurity data. In Sect. 3, we discuss various artificial neural networks and deep learning methods and their applicability within the area of cybersecurity. Several research issues and potential solutions based on our study are highlighted in Sect. 4. Finally, we conclude this paper in Sect. 5.

Understanding Cybersecurity Data

The data-driven model based on ANN and DL methods is usually based on data availability [20]. Usually, datasets reflect a series of data records consisting of many attributes or characteristics and relevant information from which the data-driven cybersecurity model is originated. In the field of cybersecurity, many datasets exist, including intrusion analysis, malware analysis, and spam analysis, which are used for different purposes. In our earlier paper “cybersecurity data science”, Sarker et al. [9], we have summarized various

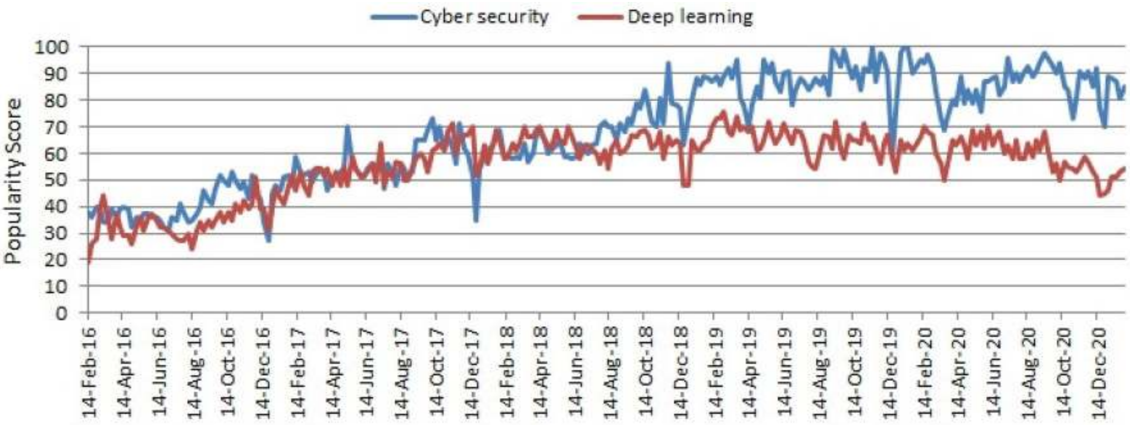

Fig. 1 The worldwide popularity score of “Cyber security” and “Deep learning” in a range of 0 (min) to 100 (max) over time where *x*-axis represents the timestamp information and *y*-axis represents the corresponding popularity score

security datasets that are obtained from different sources. In the following, several such datasets, including their different characteristics and attacks, are summarized to discuss the applicability of security modeling based on ANN and DL, according to the objective stated in this paper.

To build an intrusion detection system dataset DARPA (Defence Advanced Research Project Agency) made the earliest attempt in 1998 [21]. Under the leadership of DARPA and AFRL/SNHS, the datasets are compiled and released by the MIT Lincoln Laboratory's Cyber Infrastructure and Technology Division (formerly the DARPA Intrusion Detection Assessment Group) for the evaluation of computer network intrusion detection systems. The KDD Cup 99 dataset containing network traffic records that include more than forty feature attributes and one class identifier, is one of the most commonly used datasets for intrusion detection. [22]. The dataset contains different types of attacks that fall into four families: DoS, R2L, U2R, and PROB, as well as normal data. A refined version of this dataset is known as the NSL-KDD dataset containing similar features [23], where duplicate records are excluded from both the training and test results. As an example of security data, in Table 1, we have shown the features of intrusion detection datasets including the features and their various types such as integer, float, or nominal for a deeper understanding of security data [24]. Effectively processing these features according to the requirements, building target ANN and DL model, and eventually the decision analysis, could play a significant role to provide intelligent cybersecurity services that are discussed briefly in Sect. 3.

Another dataset the ISCX [25] was created at the Canadian Institute for Cybersecurity. To describe attack and distribution strategies in a network context, the definition of profiles was used. To create accurate profiles of attacks and other events to test intrusion detection systems, several real traces were analyzed. A new dataset, CSE-CIC-IDS2018 dataset [26], collected by the Canadian Cyber Security Institute, was recently created at the same institution, based on a user profile that tracks network events and activity. The MAWI [27] dataset is a collection of research institutions and academic institutions used by the Japanese network to calculate the global internet situation across a wide region. To track new traffic, the dataset is updated daily. For DDoS intrusion detection, some scholars use this data set [27]. The types of attacks found in it are variable since MAWI is real data traffic. The ADFA data set is a set of host-level intrusion detection system data sets issued by [28] by the Australian Security Academy (ADFA), which is commonly used in the testing of products for intrusion detection. It includes five types of attacks, including Hydra-FTP, Hydra-SSH, Add Consumer, Java-MeterPerter, Webshell, and two types of regular attacks, such as Training and Validation.

The CAIDA'07 [29], dataset represents anonymized traces of 1-h DDoS attack traffic collected on August 04, 2007. The 1-h traffic will be broken down into 5-min files.

Table 1 An example of features of an intrusion detection dataset [24]

Feature name	Value type	Feature name	Value type
dst_host_srv_count	Integer	*same_srv_rate*	Float
flag	Nominal	*dst_host_same_srv_rate*	Float
srv_serror_rate	Float	*dst_host_srv_serror_rate*	Float
dst_host_serror_rate	Float	*count*	Integer
protocol_type	Nominal	*logged_in*	Integer
dst_host_same_src_port_rate	Float	*dst_host_srv_diff_host_rate*	Float
rerror_rate	Float	*src_bytes*	Integer
dst_host_srv_rerror_rate	Float	*service*	Nominal
srv_rerror_rate	Float	*dst_host_rerror_rate*	Float
dst_host_count	Integer	*dst_host_diff_srv_rate*	Float
srv_count	Integer	*wrong_fragment*	Integer
serror_rate	Float	*num_compromised*	Integer
srv_diff_host_rate	Float	*dst_bytes*	Integer
hot	Integer	*diff_srv_rate*	Float
duration	Integer	*is_guest_login*	Integer
root_shell	Integer	*land*	Integer
urgent	Integer	*num_failed_logins*	Integer
su_attempted	Integer	*num_root*	Integer
num_file_creations	Integer	*num_shells*	Integer
num_access_files	Integer	*num_outbound_cmds*	Integer
is_host_login	Integer	–	–

The assault consists primarily of SYN, ICMP, and HTTP flood traffic. As most of the legitimate content was removed after collecting the traffic, this dataset is more biased towards DDoS attacks. The CAIDA'08 [30] dataset is the valid and attack traces tracked by Equinix (Chicago and San Jose data centers). On March 19, 2008 and July 17, 2008, respectively, traces were taken in Chicago and San Jose. The ISOT'10 dataset is a mixture of malicious and non-malicious datasets generated at the University of Victoria [31] by research in Information Security and Object Technology (ISOT). Honeynet [32] gathered decentralized botnet data for malicious traffic, and the Ericsson Research Laboratory and Lawrence Berkeley National Lab retrieved non-malicious traffic. ISCX'12 reflects the traffic from a physical test environment in the real world that produces network traffic while containing centralized botnets. A botnet traffic registered at the University of CTU, Czech Republic, in 2011, known as the [33] CTU-13 dataset. As a source of benign domain names, the Alexa Top Sites [34] dataset is commonly used as one can get as many as one million domain names. OSINT [35] and DGArchive [36] are the malicious domain names. The UNSW-NB15 dataset [37] was established in 2015 at the University of New South Wales. It has 49 characteristics and a total of almost 257,700 documents covering nine different kinds of modern attacks. A systematic approach to generate benchmark datasets for intrusion detection has been presented in [38].

In recent years, a well-organized market involving large amounts of money has become the malware industry. Top apps in the Google Play Store [39] are the most common source of normal knowledge in malware experiments. While these apps are not guaranteed to be malware-free, they are the most likely to be malware-free because of the combination of Google's vetting and the ubiquity of the apps. In addition, they are also vetted using the VirusTotal service, [40]. Malware is stored in many datasets. The Genome Project dataset [41], for example, consists of 2123 apps, 1260 of which are malicious covering 49 separate families of malware. This is similar to the Virus Share [42] and VirusTotal [40] datasets. Another wide dataset containing 22,500 malicious and 22,500 benign raw files is the Comodo dataset [43]. The Contagio [44] dataset contains 250 malicious files and is slightly smaller than the others. The DREBIN Dataset [45] is a highly imbalanced dataset containing 120,000 Android apps, 5000 of which are malicious. For the Kaggle competition, the Microsoft [46] dataset comprises 10,868 hexadecimal and assembly representation binary malware files named from nine different malware families. There are some correlations in the datasets containing malicious data and the Google Play Store data, according to the statistical details in [47] listed above. In addition, there was a broad synthetic dataset called the Computer Emergency Readiness Team (CERT) Insider Threat Dataset v6.2 [48] [49] for insider threat identification. This dataset includes 516-day device logs containing over 130 million incidents, approximately 400 of which are malicious. Due to privacy issues, email datasets are hard to obtain because they are extremely difficult to access. Some common e-mail corporations, however, include EnronSpam [50], SpamAssassin [51], and LingSpam [52]. Bot-IoT is a recent [53] dataset that includes valid and simulated IoT network traffic along with various types of forensic network analytics attacks in the Internet of Things region.

To examine the different trends of security incidents or malicious behavior, the above-discussed datasets could be used to construct a data-driven security model based on artificial neural networks and deep learning techniques. In Sect. 3, we discuss and review various ANN and DL methods by taking into account their applicability in various cybersecurity tasks.

ANN and Deep Learning in Cybersecurity

Deep learning (DL) is typically considered as a part of a broader family of machine learning methods as well as artificial intelligence (AI), which is originated from artificial neural network (ANN) [9]. The main advantage of deep learning over traditional machine learning methods is its better performance in several cases, particularly learning from large amounts of security datasets [19]. In the following, we discuss ten popular neural network and deep learning techniques including supervised, semi-supervised, unsupervised, and reinforcement learning in the context of cybersecurity. These neural networks and deep learning techniques or their ensembles and hybrid security models can be used to intelligently tackle different cybersecurity issues including intrusion detection, malware analysis, security threat analysis, predicting cyberattacks or anomalies, etc.

Multi-layer Perceptron (MLP)

Multi-layer perceptron, a class of feedforward artificial neural network (ANN), is a supervised learning algorithm [54]. It is also considered as the base architecture of deep learning or deep neural networks (DNN). A typical MLP is a fully connected network, consisting of an input layer that receives the input data, an output layer to make a decision or prediction about the input signal, and one or more hidden layers between these two [55], which are considered as the true computational engine of the network, shown in Fig. 2.

Since MLPs are fully linked, each node in one layer connects at a certain weight to each node in the next layer. Several activation functions such as ReLU (Rectified Linear Unit), Tanh, Sigmoid, Softmax [54] are used that determine the output of a network. These activation functions

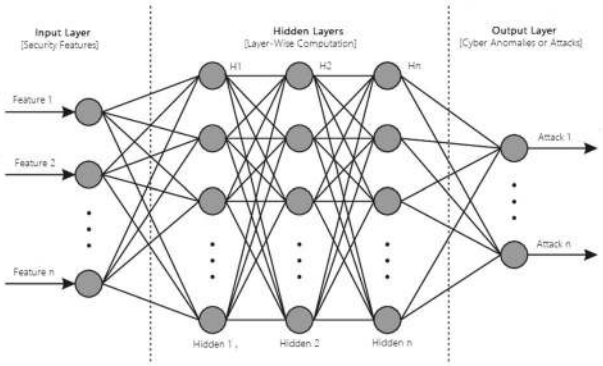

Fig. 2 An example of a feed-forward artificial neural network (ANN) with multiple hidden layers to detect cyber-anomalies or attacks

also known as transfer functions introducing non-linear properties in the network to learn complex functional mappings from data. MLP utilizes a supervised learning technique called "Backpropagation" [56] for training, which is the most "fundamental building block" in a neural network and widely used algorithm for training feedforward neural networks. The ultimate objective of the backpropagation algorithm is to optimize the network weights to accurately map the inputs to the target outputs. Various optimization techniques such as Stochastic Gradient Descent (SGD), Limited memory BFGS (L-BFGS), Adaptive Moment Estimation (Adam) [54] are used during the training process. Such neural networks can be used to solve various issues in the domain of cybersecurity. For instance, building an intrusion detection model [57], malware analysis [58], security threat analysis [59], detecting malicious botnet traffic [60] as well as for building trustworthy IoT systems [61] MLP-based networks are used. MLP is sensitive to feature scaling and needs a range of hyperparameters such as the number of hidden layers, neurons and iterations to be tuned, which may lead the model computationally expensive to solve a complex security model. However, MLP has the advantage of learning non-linear models even in real-time or on-line learning using partial fit [54].

Convolutional Neural Network (CNN or ConvNet)

The convolutional neural network (CNN or ConvNet) [62] is a deep learning network architecture that learns directly from data, without the need for manual feature extraction. A typical CNN consists of an input layer, convolutional layers, pooling layers, fully connected layers, and an output layer, as shown in Fig. 3. Thus, the CNN improves the architecture of the typical ANN, which is also considered as regularized versions of multi-layer perceptrons. Each of the layer in CNN considers optimized parameters for significant outcome as well as to reduce the complexity. CNN also uses a 'dropout' [63] that can handle the issue of over-fitting, which may cause in a typical network.

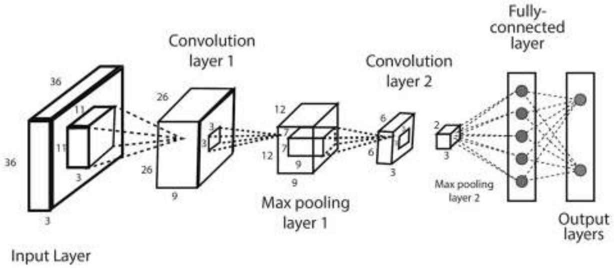

Fig. 3 An example of a convolutional neural network (CNN or ConvNet) including multiple convolution and pooling layers

Convolutional neural networks are specifically designed to deal with the variability of 2D shapes [62]. In terms of application areas, CNNs are broadly used in image and video recognition, medical image analysis, recommender systems, image classification, image segmentation, natural language processing, financial time series, etc. Although CNNs are most commonly applied to analyzing visual imagery, these networks can also be used in the domain of cybersecurity. For instance, CNN-based deep learning model is used for intrusion detection, e.g., denial-of-service (DoS) attacks, in IoT Networks [64], to detect malware [65], android malware detection [66] etc. Besides, a phishing detection model has been presented in [67] based on convolutional neural networks. A multi-CNN fusion-based model can be used for intrusion detection [68] in the area. Although CNN has a greater computational burden, it has the advantage of automatically detecting the important features without any human supervision, and thus CNN is considered to be more powerful than typical ANN. Several advanced CNN-based deep learning models, such as AlexNet [69], Xception [70], Inception [71], visual geometry group (VGG) [72], ResNet [73], etc., or other lightweight architecture of the model can be used to minimize the issues depending on the problem domain and data characteristics.

Long Short-Term Memory Recurrent Neural Network (LSTM-RNN)

Recurrent Neural Network (RNN) [74] is another type of artificial neural network, which is capable to process a sequence of inputs in deep learning and retain its state while processing the next sequence of inputs. All RNNs have feedback loops in the recurrent layer, which allows them maintaining information in 'memory' over time. Long short-term memory (LSTM) networks are a type of RNN that uses special units in addition to standard units, which can deal with the vanishing gradient problem. LSTM units have a 'memory cell' that can store data

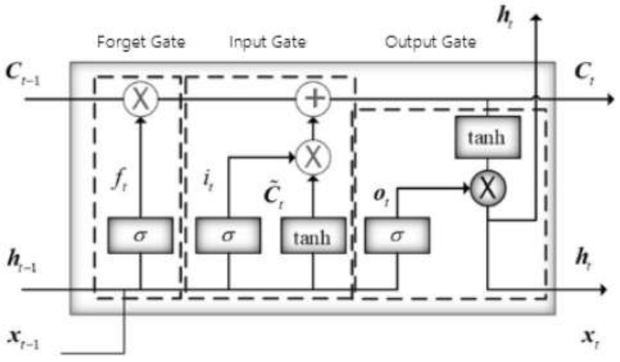

Fig. 4 Basic structure of a long short-term memory (LSTM) unit

for long periods in memory. Figure 4 shows an example of a long short-term memory (LSTM) cell, where the 'Forget Gate', 'Input Gate', and 'Output Gate' work cooperatively to control the information flow in an LSTM unit [75]. For instance, the 'Forget Gate' decides what information will be memorized from the previous state cell and to remove the information that is no longer useful, the 'Input Gate' determines which information should enter the cell state, and finally the 'Output Gate' decides and controls the outputs.

LSTM networks are well-suited for learning and analyzing sequential data, such as classifying, processing, and making predictions based on time-series data, which differentiates it from other conventional networks. Thus, LSTM is commonly applied in the area of time-series prediction, time-series anomaly detection, natural language processing, question answering chatbots, machine translation, speech recognition, etc. As a large amount of security sequential data such as network traffic flows, time-dependent malicious activities, etc. are generated these days, an LSTM model can also be applicable in the domain of cybersecurity. Several LSTM model-based security solutions such as intrusion detection [76], to detect and classify the malicious apps [77], phishing detection [78], time-based botnet detection [79] have been studied in the area. Although the main advantage of a recurrent network over a traditional network is the capability of modeling the sequence of data, it may require a lot of resources and time to get trained. Thus, considering the above-mentioned advantage, an effective LSTM-RNN network can improve the security models to detect the security threats, particularly, where the behavior patterns of the threats exhibit temporal dynamic behavior.

Self-organizing Map (SOM)

Self-organizing map (SOM) or Kohonen Map [80] is a type of artificial neural network that follows an unsupervised learning approach. It uses a competitive learning algorithm to train its network, in which nodes are competing for the right to respond to a subset of input data. It learns the shape of a dataset by continuously moving its neurons nearer to the data points. Unlike other artificial neural networks using error-correction learning such as backpropagation with gradient descent [56], SOMs implement competitive learning, a neighborhood function to preserve the topological properties of the input space. SOM is generally used for clustering [81] and mapping high-dimensional dataset as low-dimensional (typically two-dimensional) discretized pattern, which allows to reduce complex problems for easy interpretation, and thus it is known as dimensionality reduction algorithm. A Kohonen network or SOM, as shown in Fig. 5, consists of two layers of processing units called an input layer and an output layer. The units in the output layer compete with each other when an input pattern is fed to the network, and the winning output unit is typically the one whose incoming link weights are closest to the input pattern, such as measuring through Euclidean distance [56].

SOM has been widely used in, for instance, pattern recognition, health or medical diagnosis, recognition of anomalies, virus or worm attack detection [82] [83]. Several researchers have used SOM for different purposes in the domain of cybersecurity. For instance, in [84], the authors present a self-organizing map and its modeling for discovering malignant network traffic. To identify the hierarchical relations within the modern real-world datasets with mixed attributes - numerical and categorical, authors in [85] take into account the growing hierarchical self-organizing map (GHSOM) and spark-GHSOM algorithm in their analysis. The authors have shown in [86] that SOMs have a high potential as a data analytics tool on unknown traffic, where they can recognize the botnet and normal flows with high confidence of approximately 99%. SOMs are also used in [87] as a visual data mining technique while analyzing computer user behavior, security incidents, and fraud. The main

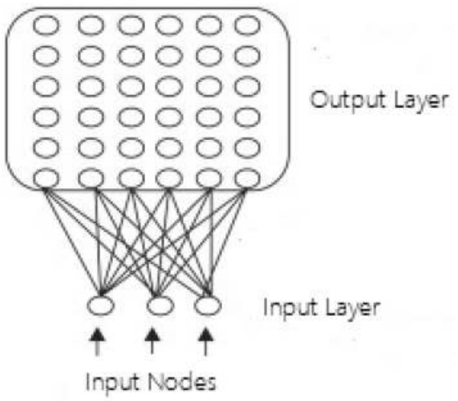

Fig. 5 The self-organizing map (SOM) architecture

advantage of using a SOM is that the data are easily interpreted and understood. Thus, SOMs can play a significant role to build a data-driven effective security model depending on the characteristics of the data.

Auto-Encoder (AE)

An auto-encoder (AE) [74] is a type of artificial neural network used in an unsupervised way to learn efficient data codes. The goal of an AE is to learn a representation for a data set, typically by training the network to ignore the 'noise' signal for dimensionality reduction. An auto-encoder consists of three components: encoder, code, and decoder as shown in Fig. 6. The encoder compresses the input and generates the data, and the decoder then uses this code to reconstruct the input. One primary benefit of the AE is that during propagation, this model can continuously extract useful features and filter the useless information [88]. A single-layered AE with a linear activation function is very similar to principal component analysis (PCA) [89], which is also used to decrease the dimensionality of large data sets.

The auto-encoder is widely used for unsupervised learning tasks, e.g., dimension reduction, feature extraction, efficient coding, and generative modeling [74, 90]. In the domain of cybersecurity, the deep AE can be used to build an effective security model. The reason is that the AE-based feature learning model in cybersecurity typically uses the minimum number of security features compared to other state-of-the-art algorithms. The resulting rich and tiny latent representation of the security features makes the model more effective and efficient, even in small devices such as smartphones, known as the internet of things (IoT) devices [91]. For example, the authors [92] present an AE-based feature learning model for cybersecurity applications, where they have demonstrated the model efficacy for malware classification and detection of network-based anomalies. An anomaly-based insider threat detection model using deep AE has been presented in [93]. In [94], the authors present a CNN-based android malware detection model, where they use deep AE as a pre-training tool to minimize the time of training. To enhance the intrusion detection method the authors in [95] use a stacked sparse auto-encoder. Thus, the AE-based model in the domain of cybersecurity can be useful due to its capability to capture the main features of data.

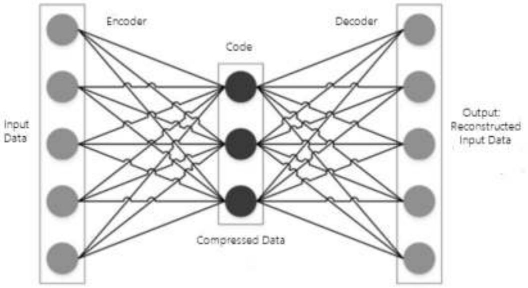

Fig. 6 A structure of an auto-encoder (AE) with the components

Restricted Boltzmann Machine (RBM)

Boltzmann machines [96] are stochastic and generative neural networks with only two types of nodes—visible nodes which we can and do measure, and hidden nodes which we cannot or do not measure. It is an unsupervised deep learning model in which every node is connected to every other node, which helps us understand abnormalities by learning about the working of the system in normal conditions. Restricted Boltzmann Boltzmann machines (RBMs) [97] are a special class of Boltzmann Machines and are limited in terms of connections between the visible layer and the hidden layer, i.e. only connections between the hidden and the visible layer of variables, but not between two variables of the same layer [96]. This restriction enables training algorithms to be more efficient than what is available for the general class of Boltzmann machines, particularly the gradient-based contrastive divergence algorithm [98]. The Figure 7 shows an illustration of an RBM consisting of m visible units $V = (v_1, ..., v_m)$ representing observable data and n hidden units $H = (h_1, ..., h_n)$ capturing dependencies between variables observed.

The RBM algorithm plays an important role in dimensionality reduction, classification, regression, collaborative filtering, feature learning, topic modeling, and many more in the era of machine learning and deep learning. In the domain of cybersecurity, the RBM can be used to build an effective security model. For example, the authors in [99] present network anomaly detection with the restricted Boltzmann machine. In their approach, they investigate the efficacy of the model to combine the expressive power of generative models with the ability to infer part of its information from incomplete training data with good classification accuracy. To increase the accuracy of DoS attack detection, the authors in [100] present a deep learning method based on a restricted Boltzmann machine. In [101], the authors present

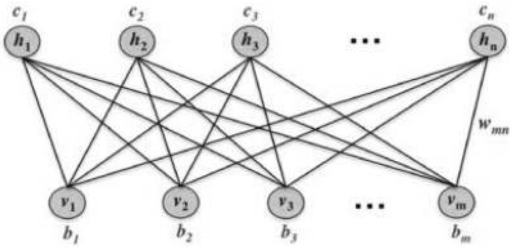

Fig. 7 A graphical representation of a restricted Boltzmann machine (RBM) with m visible and n hidden nodes

an approach for the improvement of network intrusion detection accuracy by using RBM that composes new data by removing the noises and outliers from the input data. Overall, the restricted Boltzmann machine can automatically recognize patterns in data and build probabilistic or stochastic models that incorporate randomness in the approach, which is used for feature selection and feature extraction, as well as to form a deep belief network.

Deep Belief Networks (DBN)

A deep belief network (DBN) [102] is a generative graphical model or a probabilistic generative model consists of stacked Boltzmann restricted machines (RBMs), discussed earlier. As shown in Fig. 8, it is a type of deep neural network (DNN) with multiple RBMs and a back-propagation (BP) [56] neural network. DBN can capture a hierarchical representation of input data based on its deep structure. A two-phase training can be conducted sequentially by: (1) pre-training, unsupervised layer-wise learning of stacked RBM, where the layers act as feature detectors through probabilistic reconstructing its inputs, i.e., training with the contrastive divergence [98] technique, and (2) fine-tuning, supervised learning with a classifier, e.g., BP neural network. DBN's main concept is to initialize the feed-forward neural networks with unlabeled data with unsupervised pre-training and then fine-tune the network using labeled data. DBNs can be seen as a composition of simple, unsupervised networks such as Boltzmann restricted machines (RBMs) or auto-encoders, where each sub-hidden network's layer serves as the next visible layer [103].

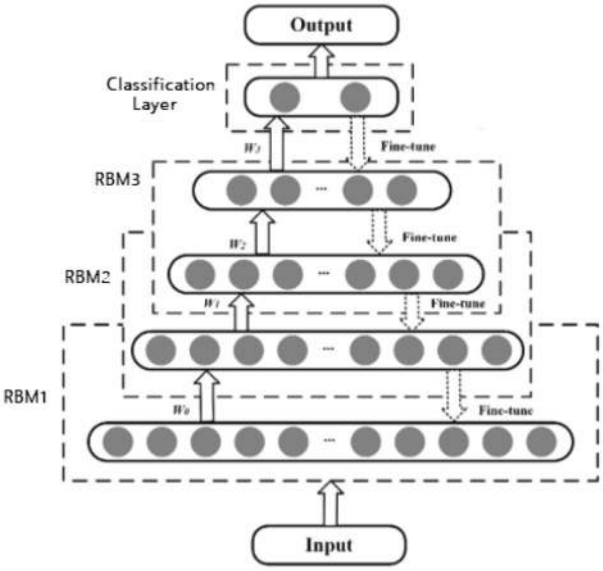

Fig. 8 Schematic structure of a deep belief network (DBN) with several layers

In the area of cybersecurity, DBN can be used in a large number of high-dimensional data applications. For instance, the authors in [104] used the DBN model as a feature reduction method to build an effective cybersecurity model, e.g., intrusion detection scheme. In [105], an intrusion detection model based on a deep belief network has been presented. Their experimental findings on NSL-KDD datasets show that there are better classification results than SVM in the DBN-based intrusion detection model, and the time of model establishment is also shorter, which significantly improves the speed of intrusion detection. The authors present an optimization technique for intrusion detection classification model based on a deep belief network in [103], where they find higher detection speed and accuracy of detection. Overall, the DBN security model can play a significant role, due to its strong capability of feature extraction and classification in a large number of high-dimensional data applications in the area of cybersecurity.

Generative Adversarial Network (GAN)

A generative adversarial network (GAN) is a class of machine learning frameworks designed by Ian Goodfellow [106], which is considered as one of the most interesting ideas in the area. Generative adversarial networks consist of an overall structure composed of two neural networks, a generator *G* and a discriminator *D*, as shown in Fig. 9, where the generator and discriminator are trained to compete with each other. The role of the generator is to generate new data with characteristics close to the actual data input. On the other hand, the discriminator is trained to estimate the probability of a future sample coming from the actual data rather than being provided by the generator.

GANs are used widely in natural image synthesis, medical image analysis, bioinformatics, data augmentation tasks, video generation, voice generation, etc. It is also useful in the domain of cybersecurity. Hackers may use an adversarial attack to access and manipulate user data in the modern world, so it is necessary to implement advanced security measures to avoid leakage and misuse of sensitive information. GAN can, therefore, be trained to recognize such cases of fraud and make deep learning models more robust.

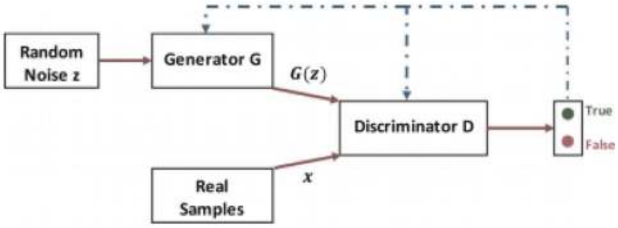

Fig. 9 Schematic structure of a generative adversarial network (GAN)

Several works have been done in the domain of cybersecurity. The authors of [107], for instance, present a transferred generative adversarial network (tGAN) for automatic zero-day attack classification and detection, which is the best performer compared to traditional machine learning algorithms. The authors present a zero-day malware detection strategy in [108] using deep auto-encoders-based transmitted generative adversarial networks, which generates fake malware and learns to distinguish it from real malware. They achieve 95.74% average classification accuracy in their experimental study. In [109], a system based on generative adversarial networks to increase botnet detection models (Bot-GAN) was presented, which improves detection efficiency and decreases the false positive rate. A new GAN-based adversarial-example attack method was implemented in [110], which outperforms the state-of-the-art method by 247.68%. In [111], the authors explore generative adversarial networks (GANs) to improve the training and ultimately performance of cyber attack detection systems by balancing data sets with the generated data. The model generates data that closely mimics the distribution of data from various types of attacks and is used to balance previously unbalanced databases, which is a viable solution for designing cyberattack intrusion detection systems. It is useful not only for unsupervised learning but also for semi-supervised learning, fully supervised learning, and reinforcement learning, depending on the task, as the main objective of GANs is to learn from a collection of training data and generate new data with the same characteristics as the training data.

Deep Transfer Learning (DTL or Deep TL)

In machine and deep learning, transfer learning is an important method for solving the fundamental problem of inadequate training data. Thus, it eliminates the need to train AI models, because it allows training neural networks with relatively small amounts of data [112]. In the field of data science, it is currently very common since most real-world problems generally do not have millions of tagged data points to train such complex models. It uses pre-trained models learned from a source domain and uses these models, shown in Fig. 10, for tasks in the target domain. Transfer learning can be classified under three sub-settings [113] based on various circumstances between the source and target domains and tasks:

- *Inductive transfer learning* In this setting, the target task varies from the source task. Several approaches such as instance transfer, feature representation transfer, parameter transfer, and relational knowledge transfer are relevant to this.
- *Transductive transfer learning* In this setting, the source and target tasks are the same, while the source and target domains are different. Several approaches such as instance transfer and feature representation transfer are relevant to this.
- *Unsupervised transfer learning* It is similar to inductive transfer learning mentioned above, where the target task is different from the source task but related to each other. It is typically studied in the context of the feature representation transfer case.

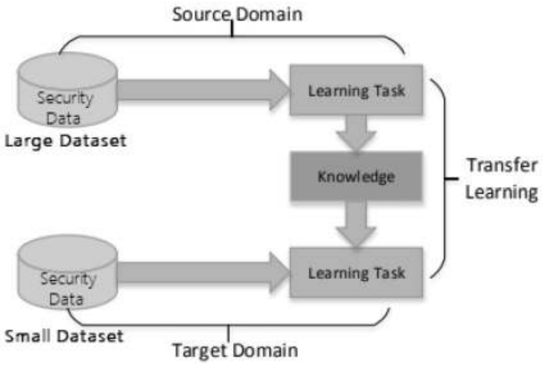

Fig. 10 Learning process of transfer learning

Deep transfer learning is applicable in various application areas such as natural language processing (NLP), sentiment classification, computer vision, image classification, speech recognition, medical imaging and spam filtering, etc. In the domain of cybersecurity, it also plays an important role due to its various advantages in modeling like saving training time, improving the accuracy of output, and the need for lesser training data. For instance, the authors in [114] present a ConvNet model using transfer learning for network intrusion detection. In [115], the authors propose a signature generation method based on deep feature transfer learning that dramatically reduces signature generation and distribution time. A higher classification accuracy of 99.5% has been achieved in [116]. The authors addressed transfer learning for the identification of unknown network attacks in [117], where they present a feature-based transfer learning approach using a linear transformation. A semi-supervised transfer learning model for malware detection is discussed in [118], where the transfer variable has improved the byte classifier accuracy from 94.72 to 96.90%. The authors present the classification of malicious software in [119], using deep neural network resnet-50 transfer learning. Their experimental findings on a sample indicate the efficacy of 98.62% accuracy in classifying malware groups. In [120] the authors present deep transfer learning for IoT attack detection with significant accuracy compared to the baseline deep learning technique. Overall, the transfer learning system significantly accelerates the training of very deep neural networks while retaining high efficiency in the field of cybersecurity, even on smaller datasets. Thus, instead of training the neural network from scratch, cybersecurity professionals can take into account a pre-trained, open-source deep learning model and finetune it for their purpose.

Deep Reinforcement Learning (DRL or Deep RL)

Deep reinforcement learning (DRL or deep RL) [135] is a category of machine learning and AI, where intelligent machines can learn from their actions similar to the way humans learn from experience. It incorporates reinforcement learning (RL) algorithms like Q-learning and deep learning, e.g., neural network learning, as defined below.

- *Reinforcement learning (RL)*—is the task of learning how agents in an environment can take sequences of actions to maximize cumulative rewards. RL considers the issue of learning to make decisions by trial and error by a computational agent.
- *Deep learning*—is a form of machine learning that uses multiple layers to progressively extract higher-level features from the raw input, and make intelligent decisions through neural network learning.

Deep RL thus incorporates deep learning models, e.g. deep neural network (DNN), based on the Markov decision process (MDP) principle [131], as policy and/or value function approximators. An MDP is "a tuple S, A, T, R, where S is a set of states, A is a set of actions, T is a mapping defining the transition probabilities from every state-action pair to every possible new state, and R is a reward function which associates a real value (reward) to every state-action pair". Figure 11 provides an example of a deep RL schematic structure. The learning system aims to allow the agent to learn to produce an optimized series of actions that maximize the total amount of rewards.

Deep RL can be used in the domain of cybersecurity. For instance, the authors in [131] demonstrate that deep RL models using deep Q-network (DQN), and double deep Q-network (DDQN) give significant intrusion detection results comparing with traditional machine learning models. Similarly, a deep RL-based adaptive intrusion detection framework based on deep-Q-network (DQN) for cloud infrastructure has been presented in [132], where they experimentally reported higher accuracy and low false-positive rates to detect and identify new and complex attacks.

Based on our study above, we have summarized the key points of each neural network and deep learning technique in Table 2. In Table 3, we have also summarized several cybersecurity applications based on these techniques. Moreover, the hybrid network model, e.g., the ensemble of networks, can be used to build an effective model considering their combined advantages. For instance, an LSTM network with the combination of CNN can also be used for detecting cyber-attacks, such as for malware detection [65], to detect and mitigate phishing and Botnet attack across multiple IoT devices [136]. Thus, we can conclude that various artificial neural network and deep learning techniques discussed above, and their variants, or modified approaches can play a significant role to meet the current needs within the context of cybersecurity.

Challenges and Research Directions

Our study on ANN and DL-based security analytics opens several research issues in the area of cybersecurity. Thus, in this section, we summarize and discuss the challenges faced and the potential research opportunities and future directions to make the networks and systems secured, automated, and intelligent.

In general, the effectiveness and the efficiency of an ANN and DL-based security solution depend on the nature and characteristics of the security data, and the performance of the learning algorithms. To collect the security data in the domain of cybersecurity is not straight forward. The current cyberspace enables the production of a huge amount of data with very high frequency from different domains. Thus, to collect useful data for the target applications, e.g., security in smart city applications, and their management is important to further analysis. Therefore, a more in-depth investigation of data collection methods is needed while working on cybersecurity data. The historical security data, discussed in Sect. 2 may contain many ambiguous values, missing values, outliers, and meaningless data. The ANN and DL algorithms including supervised, unsupervised, and reinforcement learning, discussed in Sect. 3 highly impact on data quality, and availability for training, and consequently on the security model. Thus, to accurately clean and pre-process the diverse security data collected from diverse sources is a challenging task. Therefore, existing pre-processing methods or to propose new data preparation techniques are required to effectively use the learning algorithms in the domain of cybersecurity.

To analyze the data and extract insights, there exists many neural networks and deep learning algorithms for building a security model, discussed briefly in Sect. 3. Thus, selecting a proper learning algorithm that is suitable for the target application in the context of cybersecurity is challenging. The reason is that the outcome of different ANN and DL learning algorithms may vary depending on the data characteristics [137]. We have also summarized several key points of these techniques in Table 2. Selecting a wrong learning algorithm would result in producing unexpected outcomes that may lead to loss of effort, as well as the model's effectiveness and accuracy. In terms of model building, the techniques

Table 2 A summary of artificial neural network (ANN) and deep learning (DL) networks highlighting the key points

ANN and DL techniques	Descriptive key points
Multi-layer perceptron (MLP)	Supervised learning algorithm A feed-forward fully connected artificial neural network Computationally expensive to solve a complex problem
Convolutional neural network (CNN, or ConvNet)	Regularized version of multi-layer perceptrons Can automatically learn or detect the key features from data Typically deal with the variability of 2D shapes, e.g., image
Long short-term memory recurrent neural network (LSTM-RNN)	Well-suited for learning and analyzing the sequential data Preferred for NLP tasks, speech processing, and making predictions based on time-series data
Self-organizing map (SOM)	Follows an unsupervised learning approach A dimensionality reduction algorithm used for clustering and mapping high-dimensional dataset as low-dimensional Use competitive learning rather than backpropagation
Auto-encoder (AE)	An unsupervised learning algorithm that learns a representation ofthe inputs and is deterministic To significantly reduce the noise in the input data Used typically for dimensionality reduction, very similar to PCA
Restricted Boltzmann machine (RBM)	An unsupervised learning algorithm that learns the statistical distribution and is probabilistic or stochastic Used for feature selection and feature extraction Constitute the building blocks of deep-belief networks
Deep belief networks (DBN)	A probabilistic generative model with multiple RBMs The ability to encode richer and higher order network structures and can work in either an unsupervised or a supervised setting Can be used in a large number of high-dimensional data applications
Generative adversarial network (GAN)	A form of generative model typically used for unsupervised learning Generate new, synthetic instances of data with characteristics close to the actual data input To make the deep learning models more robust
Deep transfer learning (DTL or deep TL)	To solve the basic problem of insufficient training data Use the pre-trained model and knowledge is transferred from one model to another Various advantages in modeling like saving training time, improving the accuracy of output, and the need for lesser training data
Deep reinforcement learning (DRL)	Follow the way how humans learn from experience Combines reinforcement learning (RL) algorithms like Q-learning and deep learning Can be used to solve very complex problems that cannot be solved by conventional techniques

discussed in Sect. 3 can directly be used to solve many security issues. However, the hybrid network model, e.g., the ensemble of networks, or modifying with an improvement, designing new methods, combining with machine learning techniques [138] [137] according to the target outcome could be a potential future work in the area.

Similarly, the irrelevant security data and features may lead to garbage processing as well as incorrect results, which is also an important issue in the area. If the security data is bad, such as non-representative, poor-quality, irrelevant features, or insufficient quantity for training, then the deep learning security models may become useless or will produce lower accuracy. Thus relevant and quality security data is important for better outcome. In addition to the security features, the broader contextual information [139] [140] [141] such as temporal context, spatial context, or the relationship or dependency among the events or network connections, users might help to build an adaptive system. The concept of recent pattern-based analysis, i.e., recency [142] and designing corresponding learning technique in cybersecurity solutions could also be effective depending on the problem domain. Overall, we can conclude that the success of a data-driven security solution depends on both

Table 3 A summary of cybersecurity tasks based on artificial neural network (ANN) and deep learning (DL) techniques

Used techniques	Cybersecurity tasks	References
Multi-layer perceptron (MLP)	intrusion detection, malware analysis, detecting botnet traffic, security threat analysis	Florencio et al. [57], Karbab et al. [58], Javed et al. [60], Hodo et al. [59]
Convolutional Neural Network (CNN, or ConvNet)	intrusion detection, malware detection, phishing detection, malicious user detection	Susilo et al. [64], Li et al. [68] Yan et al. [65], Mclaughlin et al. [66], Xiao et al. [67], Adebowale et al. [78], Hong et al. [121]
Long Short-Term Memory Recurrent Neural Network (LSTM-RNN)	intrusion detection, malicious activity detection, phishing detection, time-based botnet detection, authentication modeling	Kim et al. [76], Vinayakumar et al. [77], Adebowale et al. [78], Li et al. [122], Tran et al. [79], Shi et al. [123], Abuhamad et al. [124]
Self-organizing Map (SOM)	discovering malignant network traffic, modern botnets analysis, distributed clustering	Langin et al. [84], Le et al. [86] , Malondkar et al. [85]
Auto Encoder (AE)	feature learning model, insider threat detection, malware detection, intrusion detection system	Yousefi et al. [92], Liu et al. [93], Wang et al. [94], Yan et al. [95]
Restricted Boltzmann Machine (RBM)	network anomaly detection, DoS attack detection, intrusion detection	Fiore et al. [99], Imamverdiyev et al. [100], Mayuranathan et al. [125], Alom et al. [126]
Deep Belief Networks (DBN)	intrusion detection system and optimization, phishing detection, malware detection	Salama et al. [104], Qu et al. [105], Wei et al. [103], Yi et al. [127], Arshey et al. [128], Saif et al. [129], Hou et al. [130]
Generative Adversarial Network (GAN)	zero-day malware detection, botnet detection, intrusion detection systems	Kim et al. [108], Li et al. [110], Yin et al. [109], Merino et al. [111]
Deep Transfer Learning (DTL or Deep TL)	intrusion detection system, detecting unknown network attacks, malware detection, malicious software classification	Wu et al. [114], Zhao et al. [117], Gao et al. [118], Rezende et al. [119]
Deep Reinforcement Learning (DRL or deep RL)	intrusion detection system, malware detection, Security and Privacy	Lopez et al. [131], Sethi et al. [132], Fang et al. [133], Shakeel et al. [134]

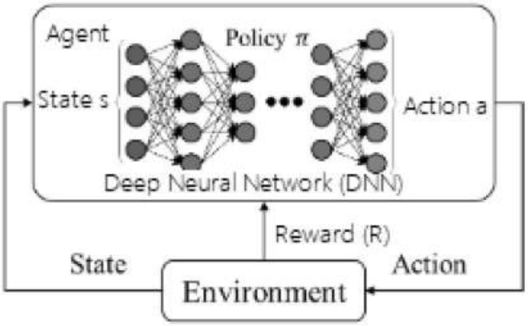

Fig. 11 Schematic structure of deep reinforcement learning (DRL or deep RL)

the quality of the security data and the performance of the learning algorithms.

Concluding Remarks

In this paper, we have conducted a comprehensive overview of cybersecurity from the perspective of artificial neural networks and deep learning methods. We have also reviewed the recent studies in each category of the neural networks to make the position of this paper. Thus, according to our goal, we have briefly discussed how various types of neural networks and deep learning methods can be used for cybersecurity solutions in various conditions. A successful security model must possess the relevant deep learning modeling depending on the data characteristics. The sophisticated learning algorithms then need to be trained through the collected security data and knowledge related to the target application before the system can assist with intelligent decision making.

Finally, we have summarized and discussed the challenges faced and the potential research opportunities and future directions in the area. Therefore, to enhance the security with time and growing popularity, the challenges that are identified create promising research opportunities in the field which must be addressed with effective solutions. Overall, we believe that our study on neural networks and deep learning-based security analytics opens a promising direction and can be used as a reference guide for potential research and applications for both the academia and industry professionals in the domain of cybersecurity.

Declarations

Conflict of interest The author declares no conflict of interest.

References

1. Li S, Da LX, Zhao S. The internet of things: a survey. Inf Syst Front. 2015;17(2):243–59.
2. McIntosh T, Jang-Jaccard J, Watters P, Susnjak T. The inadequacy of entropy-based ransomware detection. In: International conference on neural information processing. Springer; 2019. pp. 181–189.
3. Alazab M, Venkatraman S, Watters P, Alazab M et al. Zero-day malware detection based on supervised learning algorithms of API call signatures. 2010.
4. Sun N, Zhang J, Rimba P, Gao S, Zhang LY, Xiang Y. Data-driven cybersecurity incident prediction: a survey. IEEE Commun Surv Tutor. 2018;21(2):1744–72.
5. Abraham S. Data breach: from notification to prevention using PCI DSS. Colum JL Soc Probs. 2009;43:517.
6. Brij BG, Aakanksha T, Ankit KJ, Dharma PA. Fighting against phishing attacks: state of the art and future challenges. Neural Comput Appl. 2017;28(12):3629–54.
7. Ibm security report. https://www.ibm.com/security/data-breach. Accessed 20 Oct 2019.
8. Fischer EA. Cybersecurity issues and challenges: In brief. 2014.
9. Sarker IH, Kayes ASM, Badsha S, Alqahtani H, Watters P, Ng A. Cybersecurity data science: an overview from machine learning perspective. J Big Data. 2020;7(1):1–29.
10. Steven A. Cybersecurity: the cold war online. Nature. 2017;547(7661):30.
11. Anwar S, Mohamad Zain J, Zolkipli MF, Inayat Z, Khan S, Anthony B, Chang V. From intrusion detection to an intrusion response system: fundamentals, requirements, and future directions. Algorithms. 2017;10(2):39.
12. Sara M, Hamid M, Mostafa G-A, Hadis K. Cyber intrusion detection by combined feature selection algorithm. J Inf Secur Appl. 2019;44:80–8.
13. Tapiador JE, Orfila A, Ribagorda A, Ramos B. Key-recovery attacks on kids, a keyed anomaly detection system. IEEE Trans Depend Secure Comput. 2013;12(3):312–25.
14. Tavallaee M, Stakhanova N, Ghorbani AA. Toward credible evaluation of anomaly-based intrusion-detection methods. IEEE Trans Syst Man Cybern Part C (Appl Rev). 2010;40(5):516–24.
15. Farhad F, Peter L. Data science methodology for cybersecurity projects. arXiv preprint arXiv:1803.04219. 2018.
16. Saxe J, Sanders H. Malware data science: attack detection and attribution. 2018.
17. Ślusarczyk B. Industry 4.0: Are we ready? Pol J Manag Stud. 2018; 17.
18. Google trends. In https://trends.google.com/trends/, 2021.
19. Yang X, Lingshuang K, Zhi L, Yuling C, Yanmiao L, Hongliang Z, Mingcheng G, Haixia H, Chunhua W. Machine learning and deep learning methods for cybersecurity. IEEE Access. 2018;6:35365–81.
20. Aya R, Ahmed E. Data science: developing theoretical contributions in information systems via text analytics. J Big Data. 2020;7(1):1–26.
21. Lippmann RP, Fried DJ, Graf I, Haines JW, Kendall KR, McClung D, Weber D, Webster SE, Wyschogrod D, Cunningham RK, et al. Evaluating intrusion detection systems: the 1998 Darpa off-line intrusion detection evaluation. In: Proceedings DARPA information survivability conference and exposition. DISCEX'00, vol 2. IEEE; 2000. pp. 12–26.
22. Kdd cup 99. available online:http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html Accessed 20 Oct 2019.
23. Tavallaee M, Bagheri E, Lu W , Ghorbani AA. A detailed analysis of the KDD cup 99 data set. In: 2009 IEEE symposium on

computational intelligence for security and defense applications. IEEE; 2009, pp. 1–6.
24. Sarker IH, Abushark YB, Alsolami F, Khan AI. Intrudtree: a machine learning based cyber security intrusion detection model. Symmetry. 2020;12(5):754.
25. Canadian institute of cybersecurity, university of new brunswick, ISCX dataset. http://www.unb.ca/cic/datasets/index.html/. Accessed 20 Oct 2019.
26. CSE-CIC-IDS 2018 [online]. https://www.unb.ca/cic/ datasets/ids-2018.html/. Accessed 20 Oct 2019.
27. Xuyang J, Zheng Y, Xueqin J, Witold P. Network traffic fusion and analysis against DDOS flooding attacks with a novel reversible sketch. Inf Fusion. 2019;51:100–13.
28. Xie M, Hu J, Yu CE. Evaluating host-based anomaly detection systems: application of the frequency-based algorithms to adfa-ld. In: International conference on network and system security. Springer (2015).
29. Caida ddos attack 2007 dataset. http://www.caida.org/data/passive/ddos-20070804-dataset.xml/. Accessed 20 October 2019.
30. Caida anonymized internet traces 2008 dataset. http://www.caida.org/data/passive/passive-2008-dataset.xml/. Accessed 20 Oct 2019.
31. Isot botnet dataset. https://www.uvic.ca/engineering/ece/isot/datasets/index.php/. Accessed 20 Oct 2019.
32. The honeynet project. http://www.honeynet.org/chapters/france/. Accessed 20 Oct 2019.
33. The ctu-13 dataset. https://stratosphereips.org/category/datasets-ctu13. Accessed 20 Oct 2019.
34. Alexa top sites. https://aws.amazon.com/alexa-top-sites/. Accessed 20 Oct 2019.
35. Bambenek consulting–master feeds. http://osint.bambenekconsulting.com/feeds/. Accessed 20 October 2019.
36. Dgarchive. https://dgarchive.caad.fkie.fraunhofer.de/site/. Accessed 20 Oct 2019.
37. Moustafa N, Slay J. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 military communications and information systems conference (MilCIS). IEEE; 2015, pp. 1–6.
38. Shiravi A, Shiravi H, Tavallaee M, Ghorbani AA. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Comput Secur. 2012;31(3):357–74.
39. Google play store. available online: https://play.google.com/store/. Accessed 20 Oct 2019.
40. Virustotal. https://virustotal.com/. Accessed 20 Oct 2019.
41. Zhou Y, Jiang X. Dissecting android malware: characterization and evolution. In: 2012 IEEE symposium on security and privacy. IEEE; 2012. pp. 95–109.
42. Virusshare. http://virusshare.com/. Accessed 20 Oct 2019.
43. Comodo. https://www.comodo.com/home/internet-security/updates/vdp/database.php. Accessed 20 Oct 2019.
44. Contagio. http://contagiodump.blogspot.com/. Accessed 20 Oct 2019.
45. Kumar R, Zhang X, Ullah Khan R, Kumar J, Ahad I. Effective and explainable detection of android malware based on machine learning algorithms. In: Proceedings of the 2018 international conference on computing and artificial intelligence. ACM; 2018. pp. 35–40.
46. Microsoft malware classification (big 2015). http://arxiv.org/abs/1802.10135/. Accessed 20 Oct 2019.
47. Berman DS, Buczak AL, Chavis JS, Corbett CL. A survey of deep learning methods for cyber security. Information. 2019;10(4):122.
48. Lindauer B, Glasser J, Rosen M, Wallnau KC, Exactdata L. Generating test data for insider threat detectors. JoWUA. 2014;5(2):80–94.
49. Joshua G, Brian L. Bridging the gap: a pragmatic approach to generating insider threat data. In: 2013 IEEE security and privacy workshops, pp. 98–104. IEEE. 2013.
50. Enronspam. https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/enron-spam/. Accessed 20 Oct 2019.
51. Spamassassin. available online: http://www.spamassassin.org/publiccorpus/. Accessed 20 Oct 2019.
52. Lingspam. https://labs-repos.iit.demokritos.gr/skel/i-config/downloads/lingspampublic.tar.gz/. Accessed 20 Oct 2019.
53. Nickolaos K, Nour M, Elena S, Benjamin T. Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset. Future Gener Comput Syst. 2019;100:779–96.
54. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, *et al*. Scikit-learn: machine learning in python. J Mach Learn Res. 2011;12:2825–30.
55. Sarker IH. Ai-driven cybersecurity: an overview, security intelligence modeling and research directions. 2021.
56. Jiawei H, Jian P, Micheline K. Data mining: concepts and techniques. Amsterdam: Elsevier; 2011.
57. Felipe De AF, Edward DMO, Hendrik TM, Ricardo JPDBS, Filipe Barreto Do N, Flavio AOS. Intrusion detection via MLP neural network using an arduino embedded system. In: 2018 VIII Brazilian symposium on computing systems engineering (SBESC), pp 190–195. IEEE. 2018.
58. ElMouatez BK, Mourad D, Abdelouahid D, Djedjiga M. Maldozer: Automatic framework for android malware detection using deep learning. Digit Investig. 2018;24:S48–59.
59. Hodo E, Bellekens X, Hamilton A, Dubouilh P-L, Iorkyase E, Christos T, Robert A. Threat analysis of IoT networks using artificial neural network intrusion detection system. In: 2016 international symposium on networks, computers and communications (ISNCC). IEEE; 2016, pp. 1–6
60. Yousra J, Navid R. Multi-layer perceptron artificial neural network based IoT botnet traffic classification. In: Proceedings of the future technologies conference. Springer; 2019, pp. 973–84.
61. Iván G-M, Rajarajan M, Jaime L. Human-centric AI for trustworthy IoT systems with explainable multilayer perceptrons. IEEE Access. 2019;7:125562–74.
62. Yann LC, Léon B, Yoshua B, Patrick H. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–324.
63. Aurélien G. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems. O'Reilly Media, 2019.
64. Susilo B, Sari RF. Intrusion detection in IoT networks using deep learning algorithm. Information. 2020;11(5):279.
65. Yan J, Qi Y, Rao Q. Detecting malware with an ensemble method based on deep neural network. Secur Commun Netw. 2018; 2018.
66. McLaughlin N, Martinez del RJ, Kang BJ, Yerima S, Miller P, Sezer S, Safaei Y, Trickel E, Zhao Z, Doupé A et al. Deep android malware detection. In: Proceedings of the seventh ACM on conference on data and application security and privacy; 2017. pp. 301–308.
67. Xiao X, Zhang D , Hu G Jiang Y, Xia S. CNN-MHSA: a convolutional neural network and multi-head self-attention combined approach for detecting phishing websites. Neural Netw (2020).
68. Yanmiao L, Yingying X, Zhi L, Haixia H, Yushuo Z, Yang X, Yuefeng Z, Lizhen C. Robust detection for network intrusion of industrial IoT based on multi-CNN fusion. Measurement. 2020;154:107450.

69. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems; 2012, pp. 1097–1105.
70. Chollet F. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258. 2017.
71. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015, pp. 1–9.
72. Kaiming H, Xiangyu Z, Shaoqing R, Jian S. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell. 2015;37(9):1904–16.
73. Kaiming H, Xiangyu Z, Shaoqing R, Jian S. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778. 2016.
74. Ian G, Yoshua B, Aaron C, Yoshua B. Deep learning, vol. 1. Cambridge: MIT press Cambridge; 2016.
75. Changhui J, Yuwei C, Shuai C, Yuming B, Wei L, Wenxin T, Jun G. A mixed deep recurrent neural network for mems gyroscope noise suppressing. Electronics. 2019;8(2):181.
76. Jihyun K, Jaehyun K, Huong LTT, Howon K. Long short term memory recurrent neural network classifier for intrusion detection. In: 2016 international conference on platform technology and service (PlatCon). IEEE; 2016. pp. 1–5.
77. Vinayakumar R, Soman KP, Poornachandran P. Deep android malware detection and classification. In: 2017 International conference on advances in computing, communications and informatics (ICACCI). IEEE; 2017, pp. 1677–1683.
78. Adebowale MA, Lwin KT, Hossain MA. Intelligent phishing detection scheme using deep learning algorithms. J Enterp Inf Manag. 2020.
79. Tran D, Mac H, Tong V, Tran HA, Nguyen LG. A LSTM based framework for handling multiclass imbalance in DGA botnet detection. Neurocomputing. 2018;275:2401–13.
80. Teuvo K. The self-organizing map. Proc IEEE. 1990;78(9):1464–80.
81. Juha V, Esa A. Clustering of the self-organizing map. IEEE Trans Neural Netw. 2000;11(3):586–600.
82. Teuvo K. Essentials of the self-organizing map. Neural Netw. 2013;37:52–65.
83. Qu X, Yang L, Guo K, Ma L, Sun M, Ke M, Li M. A survey on the development of self-organizing maps for unsupervised intrusion detection. Mob Netw Appl. 2019; 1–22.
84. Langin C, Zhou H, Rahimi S, Gupta B, Zargham M, Sayeh MR. A self-organizing map and its modeling for discovering malignant network traffic. In: 2009 IEEE symposium on computational intelligence in cyber security. IEEE, 2009; pp. 122–129.
85. Ameya M, Roberto C, Iluju K, Michelangelo C, Nathalie J. Spark-GHSOM: growing hierarchical self-organizing map for large scale mixed attribute datasets. Inf Sci. 2019;496:572–91.
86. Le Duc C, Zincir-Heywood AN, Heywood MI. Data analytics on network traffic flows for botnet behaviour detection. In: 2016 IEEE symposium series on computational intelligence (SSCI), pp. 1–7. IEEE, 2016.
87. López AU, Mateo F, Navío-Marco J, Martínez-Martínez JM, Gómez-Sanchís J, Vila-Francés J, José Serrano-López A. Analysis of computer user behavior, security incidents and fraud using self-organizing maps. Comput Secur. 2019;83:38–51.
88. Liu W, Wang Z, Liu X, Zeng N, Liu Y, Alsaadi FE. A survey of deep neural network architectures and their applications. Neurocomputing. 2017;234:11–26.
89. Sarker IH, Abushark YB, Khan AI. Contextpca: Predicting context-aware smartphone apps usage based on machine learning techniques. Symmetry. 2020;12(4):499.
90. Guijuan Z, Yang L, Xiaoning J. A survey of autoencoder-based recommender systems. Front Comput Sci. 2020;14(2):430–50.
91. Sarker IH, Hoque MM, Uddin MK, Alsanoosy T. Mobile data science and intelligent apps: Concepts, AI-based modeling and research directions. Mob Netw Appl 1–19; 2020.
92. Yousefi-Azar M, Varadharajan V, Hamey L, Tupakula U. Autoencoder-based feature learning for cyber security applications. In: 2017 International joint conference on neural networks (IJCNN). IEEE; 2017. pp. 3854–3861.
93. Liu L, De Vel O, Chen C, Zhang J, Xiang Y. Anomaly-based insider threat detection using deep autoencoders. In: 2018 IEEE international conference on data mining workshops (ICDMW). IEEE, 2018, pp. 39–48.
94. Wei W, Mengxue Z, Jigang W. Effective android malware detection with a hybrid model based on deep autoencoder and convolutional neural network. J Ambient Intel Humaniz Comput. 2019;10(8):3035–43.
95. Binghao Y, Guodong H. Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system. IEEE Access. 2018;6:41238–48.
96. Memisevic R, Hinton GE. Learning to represent spatial transformations with factored higher-order Boltzmann machines. Neural Comput. 2010;22(6):1473–92.
97. Benjamin M, Kevin S, Bo C, Nando F. Inductive principles for restricted Boltzmann machine learning. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR workshop and conference proceedings; 2010, pp. 509–516.
98. Hinton GE, Osindero S, Yee-Whye T. A fast learning algorithm for deep belief nets. Neural Comput. 2006;18(7):1527–54.
99. Fiore U, Palmieri F, Castiglione A, De Santis A. Network anomaly detection with the restricted Boltzmann machine. Neurocomputing. 2013;122:13–23.
100. Yadigar I, Fargana A. Deep learning method for denial of service attack detection based on restricted Boltzmann machine. Big Data. 2018;6(2):159–69.
101. Seo S, Park S, Kim J. Improvement of network intrusion detection accuracy by using restricted boltzmann machine. In: 2016 8th international conference on computational intelligence and communication networks (CICN). IEEE; 2016. pp. 413–417.
102. Hinton GE. Deep belief networks. Scholarpedia. 2009;4(5):5947.
103. Peng W, Yufeng L, Zhen Z, Tao H, Ziyong L, Diyang L. An optimization method for intrusion detection classification model based on deep belief network. IEEE Access. 2019;7:87593–605.
104. Salama MA, Eid HF , Ramadan RA , Darwish A, Hassanien AE. Hybrid intelligent intrusion detection scheme. In: Soft computing in industrial applications. Springer; 2011, pp. 293–303.
105. Qu F, Zhang J Shao Z, Qi S. An intrusion detection model based on deep belief network. In: Proceedings of the 2017 VI international conference on network, communication and computing; 2017. pp. 97–101.
106. Ian G, Jean P-A, Mehdi M, Bing X, David W-F, Sherjil O, Aaron C, Yoshua B. Generative adversarial nets. In: Advances in neural information processing systems, pp. 2672–2680. 2014.
107. Jin-Young K, Seok-Jun B, Sung-Bae C. Malware detection using deep transferred generative adversarial networks. In: International conference on neural information processing. Springer; 2017. pp. 556–564.
108. Jin-Young K, Seok-Jun B, Sung-Bae C. Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders. Inf Sci. 2018;460:83–102.
109. Yin C, Zhu Y, Liu S , Fei J, Zhang H. An enhancing framework for botnet detection using generative adversarial networks. In:

2018 international conference on artificial intelligence and big data (ICAIBD). IEEE; 2018. pp. 228–234.
110. Heng L, ShiYao Z, Wei Y, Jiahuan L, Henry L. Adversarial-example attacks toward android malware detection system. IEEE Syst J. 2019;14(1):653–6.
111. Merino T, Stillwell M, Steele M, Coplan M, Patton J, Stoyanov A, Deng L. Expansion of cyber attack data from unbalanced datasets using generative adversarial networks. In: International conference on software engineering research, management and applications. Springer; 2019, pp. 131–145.
112. Weiss K, Khoshgoftaar TM, Wang DD. A survey of transfer learning. J Big Data. 2016;3(1):9.
113. Pan SJ, Qiang Y. A survey on transfer learning. IEEE Trans Knowl Data Eng. 2009;22(10):1345–59.
114. Wu P, Guo H, Buckland R. A transfer learning approach for network intrusion detection. In 2019 IEEE 4th international conference on big data analytics (ICBDA), pp. 281–285. IEEE (2019).
115. Daniel N, Aviad C, Nir N, Yuval E. Deep feature transfer learning for trusted and automated malware signature generation in private cloud environments. Neural Networks. 2020;124:243–57.
116. Nahmias D, Cohen A, Nissim N, Elovici Y. Trustsign: trusted malware signature generation in private clouds using deep feature transfer learning. In: 2019 international joint conference on neural networks (IJCNN). IEEE; 2019, pp. 1–8.
117. Zhao J, Shetty S, Pan JW, Kamhoua C, Kwiat K. Transfer learning for detecting unknown network attacks. EURASIP J Inf Secur. 2019;2019(1):1.
118. Xianwei G, Changzhen H, Chun S, Baoxu L, Zequn N, Hui X. Malware classification for the cloud via semi-supervised transfer learning. J Inf Secur Appl. 2020;55:102661.
119. Rezende E , Ruppert G, Carvalho T, Ramos F, De Geus P. Malicious software classification using transfer learning of resnet-50 deep neural network. In: 2017 16th IEEE international conference on machine learning and applications (ICMLA). IEEE; 2017. pp. 1011–1014.
120. Vu L, Nguyen QU, Nguyen DN, Hoang DT, Dutkiewicz E. Deep transfer learning for IoT attack detection. IEEE Access. 2020;8:107335–44.
121. Taekeun H, Chang C, Juhyun S. CNN-based malicious user detection in social networks. Concurr Comput Pract Exp. 2018;30(2):e4163.
122. Li Q, Cheng M, Wang J, Sun B. LSTM based phishing detection for big email data. IEEE Trans Big Data. 2020.
123. Shi W-C, Sun H-M. Deepbot: a time-based botnet detection with deep learning. Soft Comput. 2020.
124. Abuhamad M, Abuhmed T, Mohaisen D, Nyang D. AUToSen: Deep-learning-based implicit continuous authentication using smartphone sensors. IEEE Internet Things J. 2020;7(6):5008–20.
125. Mayuranathan M, Murugan M,Dhanakoti V. Best features based intrusion detection system by RBM model for detecting DDOS in cloud environment. J Ambient Intel Humaniz Comput 2019;1–11.
126. Alom MZ, Taha TM. Network intrusion detection for cyber security using unsupervised deep learning approaches. In: 2017 IEEE national aerospace and electronics conference (NAECON), pp 63–69. IEEE. 2017.
127. Yi P, Guan Y, Zou F, Yao Y , Wang W , Zhu T. Web phishing detection using a deep learning framework. Wirel Commun Mob Comput. 2018; 2018.
128. Arshey M, Angel VKS. An optimization-based deep belief network for the detection of phishing. Data Technol. Appl. 2020.
129. Saif D, El-Gokhy SM, Sallam E. Deep belief networks-based framework for malware detection in android systems. Alex Eng J. 2018;57(4):4049–57.
130. Shifu H, Aaron S, Yanfang Y, Lifei C. Droiddelver: an android malware detection system using deep belief network based on API call blocks. In: International conference on web-age information management. Springer; 2016. pp. 54–66.
131. Manuel L-M, Belen C, Antonio S-E. Application of deep reinforcement learning to intrusion detection for supervised problems. Expert Syst Appl. 2020;141:112963.
132. Sethi K, Kumar R, Prajapati N, Bera P. Deep reinforcement learning based intrusion detection system for cloud infrastructure. In: 2020 international conference on communication systems & networks (COMSNETS). IEEE. 2020; pp. 1–6.
133. Zhiyang F, Junfeng W, Jiaxuan G, Xuan K. Feature selection for malware detection based on reinforcement learning. IEEE Access. 2019;7:176177–87.
134. Shakeel PM, Baskar S, Dhulipala VRS, Mishra S, Jaber MM. Maintaining security and privacy in health care system using learning based deep-q-networks. J Med Syst. 2018;42(10):186.
135. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA. Deep reinforcement learning: a brief survey. IEEE Signal Process Mag. 2017;34(6):26–38.
136. Parra GDLT, Rad P, Kim-Kwang RC, Nicole B. Detecting internet of things attacks using distributed deep learning. J Netw Comput Appl.; 2020. 102662.
137. Sarker IH, Kayes ASM, Watters P. Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage. J Big Data. 2019;6(1):57.
138. Sarker IH. A machine learning based robust prediction model for real-life mobile phone data. Internet Things. 2019;5:180–93.
139. Sarker IH. Context-aware rule learning from smartphone data: survey, challenges and future directions. J Big Data. 2019;6(1):95.
140. Sarker IH, Colman A, Kabir MA, Han J. Individualized time-series segmentation for mining mobile phone user behavior. Comput J. 2018;61(3):349–68.
141. Sarker IH, Kayes ASM. ABC-ruleminer: user behavioral rule-based machine learning method for context-aware intelligent services. J Netw Comput Appl. 2020;168:102762.
142. Sarker IH, Colman A, Han J. Recencyminer: mining recency-based personalized behavior from contextual smartphone data. J Big Data. 2019;6(1):1–21.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.