Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-020-02014-x

ORIGINAL RESEARCH

A survey of neural networks usage for intrusion detection systems

Anna Drewek-Ossowicka[1] · **Mariusz Pietrołaj**[1] · **Jacek Rumiński**[1]

Received: 19 August 2019 / Accepted: 17 April 2020

Abstract
In recent years, advancements in the field of the artificial intelligence (AI) gained a huge momentum due to the worldwide appliance of this technology by the industry. One of the crucial areas of AI are neural networks (NN), which enable commercial utilization of functionalities previously not accessible by usage of computers. Intrusion detection system (IDS) presents one of the domains in which neural networks are widely tested for improving overall computer network security and data privacy. This article gives a thorough overview of recent literature regarding neural networks usage in intrusion detection system area, including surveys and new method proposals. Short tutorial descriptions of neural network architectures, intrusion detection system types and training datasets are also provided.

Keywords Neural network · Deep learning · Machine learning · Intrusion detection system

1 Introduction

Cyber security is an extremely important topic for contemporary society. Instant access to the global network expose individuals and organizations to cyber threats. For a while now, various methods as firewalls and antivirus software have been being used in order to protect both user's privacy and sensitive data (Choo 2011). Intrusion detection system (IDS) represents another important area for cyber security. IDS focuses on network traffic or particular computer environment analysis in order to identify signs related to malicious activity (Liao et al. 2013).

The recent rise of interest in the field of artificial intelligence (AI) resulted in major advancements of, among others, pattern recognition or anomaly detection mechanisms. Neural networks (NN) are a common choice for such problems and their usage is no longer held back. Mainly due to increase of available computational power. Such situation encouraged researchers to adapt NN architectures for IDS implementation or improvement (Saied et al. 2016; Kang and Kang 2016; Yin et al. 2017).

This article presents the results of a literature survey concerning neural networks usage in the cyber security area, specifically—intrusion detection systems. It is focused on reviewing literature in the context of the appliance of particular NN models in terms of intrusion detection systems. NN became an emerging area of interest in machine learning (ML) research activities, due to several breakthrough events, like success of convolution neural network proposals for ImageNet competition (Krizhevsky et al. 2012). This work also describes and compares recent NN methods, models used for defining new, refined IDS solutions, proposed in the reviewed literature.

The main contributions of this paper are the following:

- Review of the most relevant recent papers—methods proposal, surveys and tutorials for intrusion detection systems.
- The main focus of neural network appliance for IDSs. Other surveys known to authors generally focus on a wider field of machine learning.
- Solid base of knowledge for future researchers in terms of NN appliance to IDS.
- Stating and defining problems which have a challenging impact for related research.

✉ Anna Drewek-Ossowicka
anna.drewek@gmail.com

Mariusz Pietrołaj
mariusz.p0@gmail.com

Jacek Rumiński
jacek.ruminski@pg.edu.pl

[1] Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland

Published online: 12 May 2020

This paper is organized in the following way. Section 2 describes background of the presented research. The third chapter presents theoretical overview of IDS and NN related terms. Section 4 gives a summary of datasets that are used for IDSs, including custom solutions that we came across during our research. Next, the fifth part of this paper reveals the methodology of the literature review and decisions made during that process. Section 6 includes the results of our literature review, including an overview of surveys, new method proposals and other papers with categorization based on AI area and IDS focus. Section seven covers NN security. The last, eighth part presents our conclusion derived from the presented work.

2 Background

In our work we decided to focus mostly on neural network appliances for modern IDSs. Based on the conducted review and to our best knowledge, most surveys cover wide areas such as machine learning and/or data mining (Buczak and Guven 2016), not only neural networks. Additionally, some of them are older than 2015, which is our limit for searching the papers (Ahmad et al. 2009; Shah and Trivedi 2012; Vinchurkar and Reshamwala 2012). Such approach can be limited in terms of describing specific architectures or network models used for threats detection. Another important aspect is also a role of NN in particular solution as it can be used for classification or e.g. reduction of data dimension, which is proved by available hybrid IDS methods (Pandeeswari and Kumar 2016; Erfani et al. 2016; Al-Yaseen et al. 2017).

It can be spotted that neural networks are one of the most advancing technologies in terms of real-life influence. Robust usage of NN in mobile solutions, automotive, IoT, medical and military companies makes it an exciting technology, which is highly adaptable by industry. All of these have a high impact on number of analysis regarding NN appliance for security and privacy branches including IDS and network tracking tools.

Finally, due to rapid advancements in AI filed, new, more efficient algorithms and NN specifications are described (Almási et al. 2016). This is why focusing on the latest experiments is so important.

3 Intrusion detection systems and machine learning

3.1 Intrusion detection system

Intrusion detection systems are entities for auditing systems and network operations against hostile actions and policy violations (Tran et al. 2018). The IDS model was described firstly in 80s, by, among others, Denning (1987). IDSs can be divided into categories using several approaches. First two types are: network-based and host-based, depending on where the intrusive behavior may be observed. Network-based IDSs monitor and analyze network traffic and are focused on network security. Host-based IDSs identify malicious activities by monitoring processes and system events on the software environment that is related to particular computer (Camastra et al. 2013; Buczak and Guven 2016).

Another division of types of IDSs is based on the data analytics approaches, which have been used: signature-based (misuse-based), anomaly-based and hybrid. Signature-based approach analyses network packets or data from particular system (e.g. logs) in order to find signatures, patterns which are characteristic for intrusive behavior. This type of technique is significantly more effective in terms of known attacks as it leverages previously labelled data from database. Although it is characterized by being simple and effective method, it cannot recognize unknown attacks and requires frequent database updates (Liao et al. 2013; Modi et al. 2013; Lin et al. 2015; Buczak and Guven 2016).

Anomaly-based approach analyzes data in order to recognize abnormal situations, that differs from normal network and system behaviors. This kind of ability may be achieved based on previously provided data, which were used to train a particular algorithm. The described method is promising, because, in contrast to previous technique, it enables finding zero-day attacks. It also allows more robust customization for a particular system or network. The significant drawback in this case is the fact, that these kinds of techniques are characterized by a high level of false positive alarms, due to the fact that they are not based only on labelled data, but taught to recognize anomalies based on previously provided data, which may end up with finding situations that are anomalies, but not necessarily cyber security attacks (Liao et al. 2013; Camastra et al. 2013; Buczak and Guven 2016; Besharati et al. 2019).

Hybrid techniques are combinations of signature and anomaly detection. Such method is created in order to combine the advantages of both previous solutions—to minimize false alarm results and also raise detection effectiveness for known attacks (Buczak and Guven 2016).

A comprehensive review conducted by Liao et al. (2013) marks out also some additional types, like wireless-based, network behavior analysis, mixed IDS and stateful protocol analysis. Wireless-based IDS is analogous to network-based, it captures wireless traffic. Network behavior analysis system analyses network traffic to find malicious attacks with not expected traffic flows. Mixed IDSs combine multiple technologies to provide a more comprehensive and accurate intrusion detection. Stateful protocol analysis, on the other hand, is used to analyze specific states of the particular

Table 1 Summary of some of IDSs types (Liao et al. 2013)

IDS	Detection area	Host-based
		Network-based
		Wireless-based
		Network behavior analysis
		Mixed
	Detection methodology	Signature-based
		Anomaly-based
		Stateful protocol analysis

network protocol, to find potentially harmful patterns. (Liao et al. 2013) (Table 1).

Camastra et al. (2013) presents also the categorization of machine learning and soft computing (SC) approaches used for IDS modeling. Four groups of ML and SC are described: supervised learning-based approaches, unsupervised learning-based approaches, statistical modeling-based approaches and ensemble-based approaches. First approach is used for detecting attacks that are known, while unsupervised techniques works for new intrusions. Statistical modeling-based approach is used for monitoring user behavior and assessing whether it differs anyhow from the behavior defined as 'normal'. Ensemble-based approaches on the other hand, combine several models in order to improve efficiency and accuracy.

3.2 Neural networks

In literature, it is not obvious to find unambiguous artificial neural network definition (Guresen and Kayakutlu 2011). The accurate explanation is given by Haykin describing ANN as a "massively parallel combination of simple processing unit which can acquire knowledge from the environment through a learning process and store the knowledge in its connections" (Haykin 1994; Guresen and Kayakutlu 2011). In general, it can be stated, that neural networks aim to resemble inference of human brain.

Many different architectures of neural networks have been applied for the domain of intrusion detection systems. The most extensively used are described below (Veen 2016):

Multi-layer perceptron (MLP) is a feed-forward NN built from single perceptrons, which are simple computational models resembling biological neurons (Rosenblatt 1958). The network consists of at least three fully connected layers of perceptrons: input, hidden and output layer. The Fig. 1 presents a general example of such a network. Neurons inside a particular layer have no connection with each other.

Supervised training of MLP usually uses backpropagation algorithm based on the input and output examples provided to the network. The error between predicted and calculated results is back propagated to previous layers of the network,

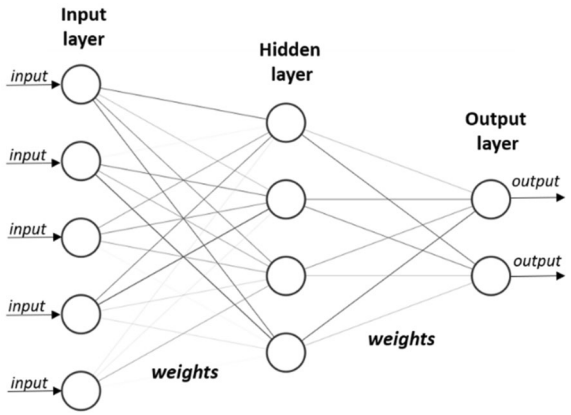

Fig. 1 The simple architecture of a three layer feed-forward neural network (LeNail 2019). Created using a program distributed with MIT license: https://github.com/alexlenail/NN-SVG and described in the article under CC-BY license: https://joss.theoj.org/papers/10.21105/joss.00747.pdf

hence the name. Figure 2 depicts a simple diagram of NN learning process including backpropagation step. With a proper number of neurons and hidden layers, MLP should be able to learn quite accurate approximation of a relation function between input and output data.

Recurrent neural network (RNN) presents an extension of standard feed-forward NN that leverages time and sequence dependencies. The main difference introduced by RNN architecture is a cyclic neuron connection, which enables inference to take into consideration previous conditions of neurons. This feature allows a network unit to remember its previous state (Elman 1990). RNN is especially useful in the area of language and video processing, where the context of data sequence is highly relevant to the structure of the input data. A major obstacle for training RNN is a known problem of gradient exploding or vanishing (Kim et al. 2016).

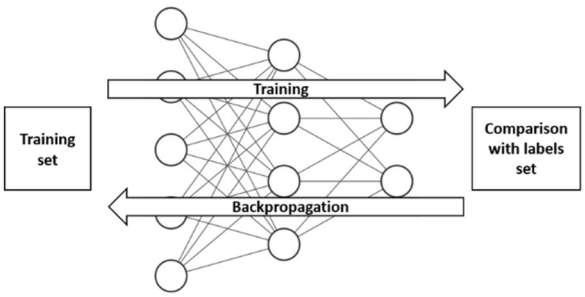

Fig. 2 The generalized learning process of artificial neural network, including feedforward and backpropagation steps (LeNail 2019). Created using a program distributed with MIT license: https://github.com/alexlenail/NN-SVG and described in the article under CC-BY license: https://joss.theoj.org/papers/10.21105/joss.00747.pdf

Long short term memory (LSTM) has been presented as a solution to difficulties related to RNN. LSTM helps to overcome the previously mentioned vanishing and the exploding gradient problem, existing in RNN. In order to avoid weight conflicts this architecture introduces a new memory cell (Hochreiter and Schmidhuber 1997). The structure of such cell includes input, output and forget gates. The main advantage of this architecture is the ability of the network to learn over long sequences of data. This is why it is widely used for text and video processing.

Autoencoder (AE) represents a variation of MLP used in an unsupervised manner, although, as present by Fig. 3., the architecture of the network is quite similar. One of the possible ways of using AE is compression or reduction of input dimensionality. Input layer processes data to output layer through limited number of hidden units, which create a bottleneck in the network structure and encode provided data (Bourlard and Kamp 1988). Decoding takes place in the further layers till the output layer, which usually corresponds with the number of neurons in the input layer. Such network construction resembles a shape of an hourglass.

Sparse autoencoder (sparse AE) is a NN, which architecture is opposite to AE presented earlier. Instead of having a bottleneck in the central part of the network, central hidden layer is the one with the highest number of neurons, which is depicted in Fig. 4 Sparse AE represents an example of an unsupervised method for learning overcomplete features. The proposed model consists of the encoder, the "sparsifying" logistics which is a non-linear data transformer, and the decoder (Ranzato et al. 2007). This architecture is mostly used in order to extricate features from a large set of unlabeled data.

Deep belief network (DBN) is an example of deep neural network, which basically consists of stacked restricted boltzmann machines (RBM) that communicate with each other. RBM is a simple two layer neural network, that can gain the

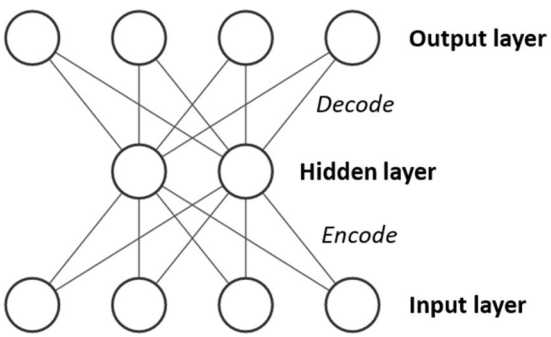

Fig. 3 The architecture of an autoencoder (LeNail 2019). Created using a program distributed with MIT license: https://github.com/alexlenail/NN-SVG and described in the article under CC-BY license: https://joss.theoj.org/papers/10.21105/joss.00747.pdf

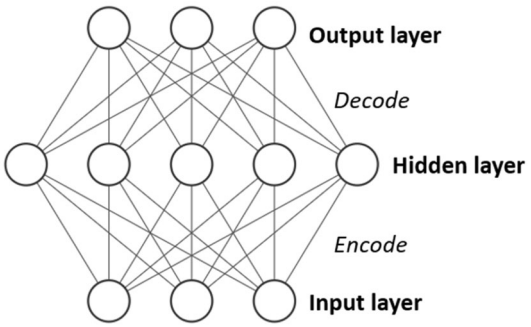

Fig. 4 The architecture of a sparse autoencoder (LeNail 2019). Created using a program distributed with MIT license: https://github.com/alexlenail/NN-SVG and described in the article under CC-BY license: https://joss.theoj.org/papers/10.21105/joss.00747.pdf

knowledge about the probability distribution of particular inputs. DBN tries to overcome the problem of not optimal solutions achieved by commonly used gradient based learning algorithms. An unsupervised greedy layer-wise learning algorithm utilized by DBN focuses on training the network part by part in order to find an optimal general solution (Liu et al. 2017).

Convolutional neural network (CNN) is a deep neural network consisted of multiple layers, as presented on Fig. 5 The main usage of CNN is image recognition, but with additional architectural or input modifications, it can be used for various other use cases. CNN in its design provides specific functions for filtering layers as convolution and pooling. In contrast to other NN architectures, not all of the layers in CNNs are fully connected. Some neurons focus on a specific group of data which helps to analyze or extract features for a particular region of an image. As these NNs are usually designed to deal with 2D shape, they are mostly used for data or image classification (Lecun et al. 1998; Guo et al. 2016).

Extreme learning machine (ELM) is an example of another modification of a standard feed-forward neural network. The main purpose of this solution is to address

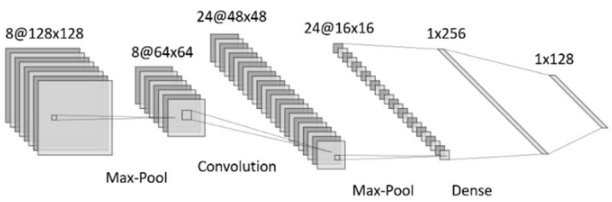

Fig. 5 An example of convolution neural network architecture (LeNail 2019). Created using a program distributed with MIT license: https://github.com/alexlenail/NN-SVG and described in the article under CC-BY license: https://joss.theoj.org/papers/10.21105/joss.00747.pdf

bottlenecks that are slowing training process and come mostly from using backpropagation based algorithms. This method can speed up the training of the network up to a thousand times with similar accuracy in comparison to standard NN methods. Such result is achieved by random connection between neuron layers and different learning algorithm based on least square fit (Huang et al. 2006).

*Self-organizing map (SOM)*defines yet another NN mechanism for unsupervised data aggregation. One of the goals stated for SOM is a reduction of dimensional complexity of input data. Due to that, such network architecture can find specific clusters of categories in a large input database. In contrast to the commonly used approaches, backpropagation is replaced here with a competitive learning algorithm to enable mapping of features (Kohonen 1982).

3.3 Other machine learning methods

Lots of research focuses on hybrid approaches to IDS, which makes NN only a part of the final method. Several different machine learning architectures are used in terms of IDS. The examples that appear in the reviewed literature are using the following other (ML) techniques:

K-nearest neighbors (K-NN) is a supervised learning algorithm based on calculating the Euclidean distance between given input data. K-NN method is commonly used for classification of given collection. The simplicity of the solution comes from categorization of the input according to calculated Euclidean distances from the classified samples from the training set. Based on that a particular element is classified by the majority of types within K-nearest neighbors, hence the name (Cunningham and Delany 2007).

*Support-vector machine (SVM)*belongs to supervised ML methods and is motivated through statistical learning theory. The process of training SVMs relates to solving a constrained quadratic optimization problem. The easiest explanation of SVM execution can be stated as finding an optimal hyperplane solution, that separates examples into two separate groups. In case of 2D space such hyperplane will be a straight line. Its main role is a binary classification of a given data. When given unlabeled data SVM can be used as a clustering mechanism (Evgeniou and Pontil 2001).

4 Datasets

Neural network training requires a significant amount of data in order to approximate effective correlation between provided input and expected results. This issue is particularly noticeable in case of supervised learning. Unfortunately, major part of publicly available datasets for IDSs is usually quite old and do not provide ideal representation of network traffic and possible threats. This obstacle might be resolved by gathering data manually or using customized versions of already available datasets. However, not having common benchmark for new IDS implementations makes it difficult to compare methods in terms of accuracy and false-positive alerts.

The following section gives an overview of datasets available for, among others, neural network training regarding IDS implementation, that were used in discussed papers and beyond. Some custom methods to generate training data are mentioned as well (Narudin et al. 2016; Wang et al. 2017) in order to help researchers find new ways to verify their own IDSs.

4.1 Public datasets

4.1.1 DARPA 1998 and DARPA 1999

The Defense Advanced Research Projects Agency (DARPA) datasets are treated as a basic, publicly available standard. DARPA 1998 was introduced by Cyber Systems and Technology Group of the Massachusetts Institute of Technology Lincoln Laboratory (Lincoln Laboratory 1998; Lippmann et al. 2000a, 2000b; Buczak and Guven 2016).

It was created based on (including both network and OS data):

- TCP/IP network data.
- Solaris basic security module logs.
- Solaris file system dumps (root and user) (Buczak and Guven 2016).

This dataset consists of network and operating system data. The data was being gathered for 9 weeks, 7 for training and 2 for testing set (Lippmann et al. 2000b).

DARPA 1999 is a successor of DARPA 1998. In this case, data was being gathered for 5 weeks, 3 for training and 2 for testing. The major distinction between them is an expanded range of available attack scenarios (Lippmann et al. 2000a).

However DARPA 1998 and DARPA 1999 are usually presented as commonly used datasets for experiments, during our research we did not encounter new methods using them. The possible reason behind it is, that those datasets turned out to not be fully capable of simulating physical network systems (McHugh 2000; Aljawarneh et al. 2018) and are currently being replaced by newer proposals.

4.1.2 KDD Cup 1999

The KDD Cup 1999 dataset (KDD Cup 1999) is one of the most often used datasets for evaluating IDSs. It utilizes TCP/IP data from DARPA 1998 dataset. While DARPA 1998

consists of about 5 million records in training data and around 2 million records in testing data, KDD Cup 1999 training part has around 4,900,000 connection vectors (Tavallaee et al. 2009). Each vector has 41 features and is classified as normal connection or an attack. Additionally it can belong to one of four attack types (Tavallaee et al. 2009; Dhanabal and Shantharajah 2015):

Denial of service (DOS)—a case when an attacker purposely uses victims resources with flood number of malicious request in order to make it unable to handle legitimate calls to the service.

User to root (U2R)—rising normal user privileges to a super user (root) by exploiting some vulnerabilities in the attacked system.

Probe (probing)—exploring or examining victim or its environment in order to gain information. Port scanning or checking duration of connection are only a few examples.

Root to local (R2L)—access of an unauthorized entity to a remote machine and gaining local privileges.

The 41 features are divided into three groups (Tavallaee et al. 2009):

Basic features—general features for TCP connections.

Content features—features describing invalid behaviors for single connections helping discovering R2L and U2R attacks.

Traffic features—features defined using time window.

Besides huge popularity and number of available data, KDD Cup 1999 struggles with problems. Some of them were inherited from DARPA'98 dataset like the fact of being fully synthetic dataset or lack of the examination of possible dropped packets while the dataset was being created. KDD Cup 1999 itself suffers also from not even distribution of the attacks and record redundancy (Tavallaee et al. 2009). While describing KDD Cup 99, we spotted, that one of the traffic features—*dst_host_same_src_port_rate*—is described in literature as "*same_src_port_rate* for destination host" (KDD Cup 1999; Shanmugavadivu and Nagarajan 2011; Amiri et al. 2011; Songma et al. 2012), while in KDD Cup 1999, to our best knowledge, we could not find *same_src_port_rate* feature, hence lack of description in Table 2.

4.1.3 NSL-KDD

NSL-KDD is a dataset that was created to overcome the issues of DARPA and KDD Cup 1999 datasets (Tavallaee et al. 2009; Dhanabal and Shantharajah 2015). It was proposed by Tavallaee et al. (2009). The main advantages over KDD Cup 1999 are (NSL-KDD 2009):

- Lack of redundant records in training set and no records in testing set, that are duplicated.
- The number of records is feasible, so there is no need of creating subsets of the dataset for the experiments.
- Inverse proportion number of particular records from each difficulty level group to the percentage of records in the original KDD Cup 1999 dataset.

NSL-KDD is still not perfect (due to problems that are going to be listed in the next section), nevertheless can be used for effective benchmarking for IDSs.

4.1.4 UNSW-NB15

Extensive usage of KDD Cup 1999 and NSL-KDD datasets resulted in discovering the following challenges:

- Missing some low footprint attack characteristics,
- Missing some traffic schemes (e.g. normal and modern),
- Discrepancy between distribution of particular data sets (training vs. testing) (Moustafa and Slay 2016).

UNSW-NB15 was created as a response to the above problems. It was created with the usage of an IXIA PerfectStorm tool in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) (UNSW-NB15 2015; Moustafa and Slay 2016). UNSW-NB15 consists of 49 features. There are two attributes for the data provided: *label* (0 for normal and 1 for otherwise) and *attack_cat* for attack category (Moustafa and Slay 2016). There are five categories of features: Flow, Basic, Content, Time and Additional Generated Features. The types of attacks are: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms (Moustafa and Slay 2015, 2016) (Table 3).

4.1.5 Kyoto2006+

Another example of publicly available benchmark data for IDS training and testing is Kyoto2006 + dataset. It presents 24 different network related features which have been extracted from servers placed at Kyoto University. 14 features are obtained from KDD Cup 99, while 10 other features were newly added (Song et al. 2011). Data was gathered for three years from 2006 to 2009 (Ambusaidi et al. 2016). It was created as an alternative for KDD Cup 1999 (Song et al. 2011). There is also benchmark version described, which contains 17 features (14 derived from KDD Cup 1999 and 3 additional) (Kyoto dataset 2015).

4.2 Other datasets

Some of the researchers decided to experiment with other than any of presented above public datasets. Erfani et al. (2016) used six real-life datasets and two synthetic ones. The six real-life ones were received from the UCI Machine Learning Repository:

Table 2 Features of individual TCP connections in KDD Cup 1999 (KDD Cup 1999; Amiri et al. 2011)

Feature	Description
Basic features	
Duration	Connection length expressed in seconds
Protocol_type	Type of connection protocol, e.g. udp
Service	Destination network service, e.g. telnet
Flag	Connection status—normal/error
Src_bytes	Bytes from source to destination point
Dst_bytes	Bytes from destination point to source
Land	1 or 0 – if connection is from the same host/port
Wrong_fragment	Number of incorrect fragments
Urgent	Number of packets marked as urgent
Content features	
Hot	"Hot" indicators—number
Num_failed_logins	Failed logins attempted—number
Logged_in	1 or 0—if login trial was successful
Num_compromised	"Compromised" conditions - number
Root_shell	1 or 0—if root shell was accessed
Su_attempted	1 or 0—if there was "su root" attempt
Num_root	"Root" accesses—number
Num_file_creations	File creation operations—number
Num_shells	Shell prompts—number
Num_access_files	Operations on access control files—number
Num_outbound_cmds	Outbound commands in ftp conn—number
Is_hot_login	1 or 0—if login is on the "hot" list
Is_guest_login	1 or 0—if the login is classified as "guest"
Traffic features	
Count	Connections as current to the same host—number in the past two seconds
Srv_count	Connections as current to the same service—number in the past two seconds
Serror_rate	Connections having "SYN" errors—percentage
Srv_serror_rate	Connections having "SYN" errors—percentage (service)
Rerror_rate	Connections having "REJ" errors—percentage
Srv_rerror_rate	Connections having "REJ" errors—percentage (service)
Same_srv_rate	Connections to the same service—percentage
Diff_srv_rate	Connections to different services—percentage
Srv_diff_host_rate	Connections to different hosts—percentage
Dst_host_count	Count for destination host
Dst_host_srv_count	srv_count for destination host
Dst_host_same_srv_rate	Same_srv_rate for destination host
Dst_host_diff_srv_rate	Diff_srv_rate for destination host
Dst_host_same_src_port_rate	Lack of detailed description
Dst_host_srv_diff_host_rate	Srv_diff_host_rate for destination host
Dst_host_serror_rate	Serror_rate for destination host
Dst_host_srv_serror_rate	Srv_serror_rate for destination host
Dst_host_rerror_rate	Rerror_rate for destination host
Dst_host_srv_rerror_rate	Srv_rerror_rate for destination host

- Forest adult gas sensor array drift (Gas).
- Opportunity activity recognition (OAR),
- Daily and sport activity (DSA),
- Human activity recognition using smartphones (HAR),

and have dimensionalities of 54, 123, 128, 242, 315 and 561 attributes. Synthetic datasets were "Banana" dataset, created by mixing "two banana shaped distributions" and

Table 3 UNSW-NB15—group of features and labels (Moustafa and Slay 2015, 2016)

Group of features	Name	Group of features	Name
Flow features	srcip	Time features	sjit
	sport		djit
	dstip		stime
	dsport		ltime
	proto		sintpkt
Basic features	state		dintpkt
	dur		tcprtt
	sbytes		synack
	dbytes		ackdat
	sttl	Additional features	is_sm_ips_ports
	dttl		ct_state_ttl
	sloss		ct_flw_http_mthd
	dloss		is_ftp_login
	service		ct_ftp_cmd
	sload		ct_srv_src
	dload		ct_srv_dst
	spkts		ct_dst_ltm
	dpkts		ct_src_ ltm
Content features	swin		ct_src_dport_ltm
	dwin		ct_dst_sport_ltm
	stcpb		ct_dst_src_ltm
	dtcpb	Labels	attack_cat
	smeansz		label
	dmeansz		
	trans_depth		
	res_bdy_len		

the second one was "Smiley"—combination of Gaussians and arc shaped distributions (Erfani et al. 2016).

Kang and Kang (2016) were working on the data created by packet generator open car test-bed and network experiments (OCTANE) (Borazjani et al. 2014), which was able to generate CAN (controller area network) packets, a standard for in-vehicle network communication.

Narudin et al. (2016) focused on mobile malware detection. They used two datasets: public (MalGenome) and self-collected, private dataset. MalGenome consists of 1260 malwares records categorized into 49 different groups. It was gathered between 2010 and 2011 (Narudin et al. 2016).

Saied et al. (2016) used artificial neural network for detecting DDoS attacks. In order to generate datasets, they built safe, realistic network, where they performed DDoS attacks (TCP, UDP and ICMP protocols).

David and Netanyahu (2015) used an extensive dataset provided by C4 Security with multiple malware categories. It was used for Deep Belief Network to generate signatures for malware records.

An interesting example is provided by Du et al. (2017) - HDFS log dataset and OpenStack log dataset. First comes from Hadoop - based environment, second from the OpenStack environment. Those datasets are particularly interesting due to be some examples of datasets based on logs, not network packets.

Wang et al. (2017) present, on the other hand, an example of self-generate dataset called USTC-TFC2016. It consists of two parts. First contains ten types of malware traffic from publicly accessible website, second contains ten types of non-malware traffic.

Network traffic data can also be gathered by NetFlow, which was created as Cisco router feature, as mentioned in (Buczak and Guven 2016). Network flow in this understanding is an order of packets sharing exactly the same packet features: IP protocol, source port, destination port, IP type of service, ingress interface, source IP address and destination IP address (Buczak and Guven 2016).

Both approaches (public vs. private/privately generated) datasets have their pros and cons. In case of public datasets it is possible to easily compare the results of the experiments with other methods results and benchmark particular solutions. On the other hand, those datasets are considered two general and not flexible enough to address contemporary needs in terms of IDSs. Private datasets can be prepared for specific experiment and better address particular needs, nevertheless they can be a subject of privacy concerns and have too specific form, that is hard to be used on a wider scale.

5 Research method

This paper is designed for researchers, who need a complex source of data concerning available literature in terms of Neural Network usage for Intrusion Detection Systems. We decided to review the newest scientific literature concerning the topic above. In order to achieve that, we performed a systematic literature review. Google Scholar database was used for performing research for two search strings: (1) "intrusion detection system" AND "neural network", (2) "intrusion detection system" AND "neural networks". We did not include the word "artificial" in the search string, due to the fact that in the articles NN are covered by both "neural network(s)" and "artificial neural network(s)" phrase, so we did not want to exclude accidentally any important articles. Especially, taking into consideration the fact, that quite frequently, "Artificial Neural Network" term is used for particular architecture, like multi-layer perceptron. These search string were searched separately, due to lack of confirmation that Google Scholar accepts any parenthesis in search strings (Tay 2015). *Publish or Perish* software (Harzing 2007) was

used to perform the queries due to the ease of exporting data to Excel files, which was necessary to perform later review. We searched for articles from between 2015 and 2019 and sorted the results based on number of citations. The search was performed on April 6th, 2019. The goal was to review the literature that currently has the biggest influence and we decided that citations count is a quite good indicator of the potential paper impact, as the authors of other surveys/literature reviews proposed (Buczak and Guven 2016). From each, sorted list (from each of two search strings) we excluded patents and books in order to focus on journal and conference papers. After exclusion we chose 50 positions from each list and merged both lists. Majority of entries were repeated so the final number of journals prepared for abstract review was 62 articles. Next, we performed an abstract review to assess if a particular article is relevant to our research. The final list of articles for literature review contained 34 articles (Fig. 6).

We are conscious that citation number is not flawless. The older the paper is the bigger chance it obviously gets to obtain high number of citations. That is why the decision was to review papers from short time range. We are also aware that survey and tutorials, due to its informative nature, are getting high number of citations in general, hence in the final summary they are presented in a separate category (Table 4).

6 Results

The reviewed articles can be categorized into three separate groups. First one consists of surveys focused on machine learning algorithms usage for IDSs. It covers not only particular examples of intrusion detection methods, but also general knowledge about Artificial Intelligence and datasets available for ML algorithms training. The second group gathers all articles that focus on new methods proposals or experiments including strict neural network usage and hybrid solutions for IDSs. The third group, which depicts articles

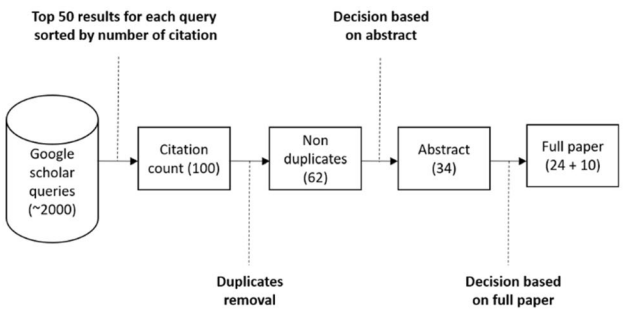

Fig. 6 Number of the articles at each of manual literature review steps

Table 4 Search criteria used for literature review

Search parameter	Value
Database	Google Scholar
Search date	06th of April 2019
Search strings	"Intrusion detection system" AND "neural network" "Intrusion detection system" AND "neural networks"
Timeline	2015–2019
Sorting	Highest citation count
Document type	Journals Conference papers

that cannot be categorized into one of two first groups. Ten articles have been marked as other as they focus mostly on datasets itself or machine learning methods, which are not strictly related to NN. For the purpose of this research we decided to include them as well as they present a wider range of available IDS solutions. The below chart summarizes ratio of the reviewed articles (FIg. 7):

6.1 Surveys

During the conducted literature review we came across couple of surveys regarding possible IDS implementations or enhancements with usage of Artificial Intelligence. Most of the articles present a good theoretical machine learning and IDSs background for researchers interested in this field. Nevertheless, neural networks are usually treated only as a small part of available solutions. It is also worth mentioning, that papers that emphasized recent advancements in deep learning were able to depict a wider area of NN field.

Having in mind how important the aspect of datasets is for NN related methods, we decided to review mentioned surveys in terms of described or proposed datasets. This shows, that besides long-serving databases as KDD Cup 1999, NSL-KDD etc. it is hard to define reliable dataset for

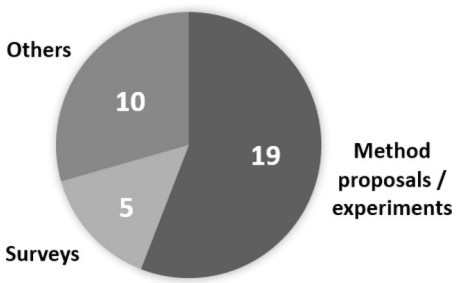

Fig. 7 The categories of the final group of the reviewed articles

Table 5 Summary of reviewed surveys

Authors	Citations (6th Apr 2019)	AI area focus	Datasets focus	Summary	Year
Buczak and Guven (2016)	423	Machine learning and data mining: Neural networks Association and fuzzy association rules, Bayesian network, Clustering, Decision trees, Ensemble learning, Evolutionary computation, Hidden Markov models, Inductive learning Naïve Bayes, vSequential pattern mining, Support vector machine	Netflow DARPA 1998 DARPA 1999 KDD Cup 99 NSL-KDD	Detailed overview of available data mining and Machine Learning methods including comparison of datasets, performance comparisons and recommendations against IDS implementation	2016
Agrawal and Agrawal (2015)	139	Clustering: k-means, k-medoids, EM clustering, Outlier detection algorithms classification: Classification tree, Fuzzy logic, Naïve Bayes network, Genetic algorithm, Neural networks, Support vector machine hybrid: Cascading supervised techniques, Combining supervised and unsupervised techniques	N/A	Overview of available Data Mining techniques and hybrid methods with examples that have been implemented for IDSs. Focused on Anomaly Detection	2015
Fadlullah et al.. (2017)	117	Machine learning and deep learning: Convolutional NN Recurrent NN Long short term memory NN Stacked auto-encoder Deep Boltzmann machines Deep reinforcement learning	N/A	Overview of deep learning architectures and their appliances for network related traffic control	2017
Narudin et al. (2016)	104	Machine learning: Random forest J48 Multi-layer perceptron Bayesian network k-NN	MalGenom Custom, self-collected database	Focus on mobile malware detection also with usage of IDS. Overview of possible appliances of described methods and their verification on chosen datasets. It is not a classic survey, rather evaluation of existing methods – authors decided to keep it in this category, as to our best understanding it provides evaluation for machine learning classifiers	2016
Kwon et al. (2019)	51	Deep learning and machine learning: Restricted Boltzmann machine Deep belief network Deep neural network Recurrent neural network	KDD Cup 99 NSL-KDD	Overview of multiple methods of data dimensionality reduction and possible DL appliances to IDS enhancement. Authors performed also an experiment with Fully Connected Network model for NSL-KDD dataset	2017

benchmarking that could be used across all reviewed solutions. UNSW-NB15 is an example of the newer one.

Table 5 briefly summarizes surveys that were finally chosen for review in our research. As mentioned before, we decided to focus on two major aspects, which are AI area and datasets described in a particular paper. This gives a good base for further data gathering or research in terms of NN and IDSs.

Not all ML techniques have been thoroughly explained in this paper. The number of proposed solutions is so high, that we decided to list them in the table and redirect our readers to a specific article.

6.2 New methods proposals and experiments

Major part of the reviewed articles presents new methods for IDSs, based on neural network or performed experiments. The below table summarizes NN architectures used by researchers. Additionally, dataset used for method validation is stated. We did not perform accuracy comparison of the proposed solutions. Such comparison might be not informative due to different datasets or data subsets that were used. Additionally there are some differences during data preparation steps or type of attacks detected by particular IDS. Due to vast variety of available methods proposal, only part of the below algorithms or NN architectures have been described in this article. For each listed publication, column "method used" enlist general mechanism used by the particular solution. For more details the reader is redirected to the related paper (Table 6).

6.3 Other related papers

Some of the papers that we reviewed could not be easily classified to the category of surveys or new method proposals/experiments. In this group we placed works that present:

- Interesting IDS enhancing methods, that are not directly connected to neural networks,
- Papers that focus on datasets itself.

However, this paper focuses on NN based IDSs, we think that mentioning most cited ML based solution might be beneficial for future research. As we presented, quite often hybrid methods are used instead of plain NN. Although those articles do not match exactly our criteria of research, we found them useful in terms of appliance in the field of IDS.

Publications focusing on comparison and analysis of the datasets might be especially helpful as number of public training data for IDS is quite limited. Extended knowledge on structure and possible challenges of these common learning sets might enable researchers to improve the accuracy of the proposed solutions (Table 7).

7 Security concerns

In this paper, we present the overview of the latest literature concerning NN usage in IDSs. While describing this topic, it is important to highlight, that IDS can be itself a subject of security attack (Corona et al. 2013). Also machine learning based solutions usage in modern IDS architecture can raise security concerns. Appliance of machine learning in cybersecurity area may result in undesirable inheritance of its flaws by NN based IDSs and new vectors of attacks.

Corona et al. (2013) provided an interesting taxonomy proposal for adversarial attacks against Intrusion Detection Systems in general. The types of attacks that can directly harm NN based IDSs are, among others poisoning and evasion (Corona et al. 2013; Pitropakis et al. 2019). The first type of attacks concerns manipulating training data in order to decrease algorithm's performance, resulting in, for example, misclassification (Baracaldo et al. 2018). This obviously concerns wide usage of ML algorithms, not only in IDSs (Baracaldo et al. 2018). Evasion attacks, on the other hand, are focused on the testing phase of the algorithm. Pitropakis et al. (2019) provide an example of such attack in the context of NN and IDSs. They describe experiment prepared by Demetrio et al. (2019), where the evasion black-box attack was performed against convolutional neural network, in order to compromise its classification possibilities (Pitropakis et al. 2019).

Another classification of attacks that can be performed on ML based IDSs is differentiation between black-box, gray-box and white-box attacks (Darvish Rouani et al. 2019). In case of black-box attacks, intruder has no knowledge about the ML algorithm or model. Gray-box attacks involves only knowledge about ML algorithm or model, but without any information about model parameters. In terms of white-box attack – the attacker has knowledge about all of the above (Darvish Rouani et al. 2019).

The most important thing is, how IDSs that use ML in general (including NN), can be defended from adversarial attacks. One of the solutions for defending from poisoning attack is training data manipulation, nevertheless it can cost increased computational resources (Corona et al. 2013). One of the proposals in the literature for NN defense in general is Mixup, which, among others, helps to act against adversarial examples (Zhang et al. 2018; Stewart 2019). Yuan et al. (2019) presented a classification of two types of defense strategies against adversarial examples: reactive and proactive. The first type consists of adversarial detecting, input reconstruction and network verification, while the second

Table 6 List of reviewed new IDS method

Authors	Citations (6th Apr 2019)	Dataset	Method used	IDS focus	Summary	Year
Erfani et al. (2016)	198	Six real-life and two synthetic datasets (not network related)	Hybrid: DBN - reduction of data dimensionality SVM – anomaly detection	N/A	New method proposal of unsupervised dimensionality reduction. No example on network traffic database has been presented.	2016
Ashfaq et al. (2017)	187	NSL-KDD	Hybrid: NN - preparation of fuzzy membership vector Fuzziness based algorithm- categorization	DOS U2R R2L PROBE	Semi-supervised ML method. Results compared with: J48, Naive Bayes, NB tree, Random forests, Random tree, Multi-layer perceptron, SVM. Two-class classification: normal vs. attack	2017
Javaid et al. (2016)	142	NSL-KDD	Hybrid: Sparse AE – unsupervised feature learning Soft-max regression - classifier	DOS U2R R2L PROBE	ML method for IDS enhancement – Self-taught learning. 2, 5 and 23 – class classification.	2016
Kang and Kang (2016)	120	Generated with OCTANE simulation software	DBN DNN	VANET communication: Vehicle to vehicle Vehicle to infrastructure	Method for threats detection in automotive communication.	2016
Tang et al. (2016)	96	NSL-KDD	Deep NN (three hidden layers)	DOS U2R R2L PROBE	Method based on six feature subset from NSL-KDD. Comparison of results with: J48, Naive Bayes, NB Tree, Random Forest, Random Tree, MLP, SVM. Focused on Software Defined Networking	2016
Saied et al. (2016)	94	Generated with simulation of DDOS attacks with special tools.	NN	DDOS	Supervised method for detection of known and unknown DDOS attacks in real time.	2016
Aljawarneh et al. (2018)	93	NSL-KDD	Hybrid: First - filtering data with Vote algorithm Second: two of classifiers: J48 Meta pagging Random tree REPTree AdaBoostM1 Decision stump Naïve Bayes	DOS U2R R2L PROBE	Method including variety of available classifiers. Clear comparison of presented mechanism in respect to possible attack types. Binary and multiclass classification.	2018
Ozay et al. (2016)	88	Generated test system for smart grid	Various machine learning techniques	Smart grid related attacks	Technical overview with performance and accuracy comparison of new ML methods for smart grid attack detection	2016

Table 6 (continued)

Authors	Citations (6th Apr 2019)	Dataset	Method used	IDS focus	Summary	Year
Yin et al. (2017)	82	NSL-KDD	Recurrent NN	DOS U2R R2L PROBE	Method including experiments with different topologies of recurrent neural network. Detailed comparison with other ML methods as : J48, Naïve Bayes, NB Tree Random Forest, MLP, SVM	2017
Singh et al. (2015)	79	NSL-KDD Kyoto 2006+	Online sequential extreme learning machine (OS-ELM)	DOS U2R R2L PROBE	Methodology with reduced computational time and memory requirements. Multiple topology of proposed neural network architecture are proposed based on manipulation of neurons count in hidden layer. Results of the proposed technique are compared with: ANN, AdaBoost, Native Bayes and ELM	2015
Kim et al. (2016)	74	KDD Cup 1999	LSTM + RNN	DOS U2R R2L PROBE	ML method with training data based on KDD Cup 99 subset. The article includes experimentation on NN parameters in order to improve the proposed solution. Results are compared with: GRNN, PNN, RBNN, KNN, SVN, Bayesian	2016
Hodo et al. (2016)	66	Internet packets	NN	DOS DDOS	Method focused on IoT security against DoS and DDoS attacks. The solution is validated based on a simulated IoT Network	2016
Pandeeswari and Kumar (2016)	62	KDD Cup 1999	Hybrid: Fuzzy means clustering (FMC) - clustering of incoming data NN - trained based on FMC output	DOS U2R R2L PROBE	Hybrid method proposed for cloud environments. Results compared with: Naïve Bayes and ANN	2016

Table 6 (continued)

Authors	Citations (6th Apr 2019)	Dataset	Method used	IDS focus	Summary	Year
De la Hoz et al. (2015)	61	NSL-KDD	Hybrid: Principal component analysis (PCA) / Fisher discriminant ratio (FDR) - feature selection and noise removal Self-organizing map - classification	DOS U2R R2L PROBE	Unsupervised hybrid method for Intrusion Detection Systems including variation of hybrid classification methods based on SOM.	2015
Du et al. (2017)	60	HDFS logs OpenStack logs VAST challenge 2011 (testing)	LSTM	Suspicious activities: DOS Port scan Socially engineered attack Undocumented IP address	Method based on system logs as a training data. Verification done also on VAST challenge 2011 dataset	2017
Shone et al. (2018)	56	KDD Cup 99 NSL-KDD	Non-symmetric deep AE Random forest	DOS U2R R2L PROBE	Method based on stacked AE and Random Forest. Extensive comparison with DBN results per each network threat defined in KDD dataset	2018
Alom et al. (2015)	50	NSL-KDD	DBN	DOS U2R R2L PROBE	Method plainly using DBN. According to authors it achieves better accuracy than SVM and DBN-SVM based solutions with only subset of NSL-KDD database used as a training input	2015
Wang, et al. (2017)	47	Self-created USTC-TFC2016	CNN	10 types of normal traffic, e.g. Gmail 10 types of malware traffic, e.g. Cridex	Method taking network traffic data as images - training input for CNN. Using raw traffic data. The paper includes new data set of network traffic created by authors called USTC-TFC2016 (around 3.71 GB) along with data preprocessing toolkit called USTC-TK2016	2017
Alheeti et al. (2015)	46	Generated with simulation tools: SUMO and MOVE	NN	VANET communication (DOS): Vehicle to vehicle Vehicle to infrastructure	IDS method for DOS detection in VANET infrastructure	2015

Table 7 Other reviewed papers

Authors	Citations (06.04.2019)	AI Area	Dataset focus	Summary	Year
Lin et al. (2015)	214	Cluster center and nearest neighbor approach (CANN). Contains k-NN	KDD Cup 99	Proposal of a new ML feature representation method based on cluster center and nearest neighbor approach	2015
Aburomman and Ibne Reaz (2016)	132	Hybrid: SVM k-NN	KDD Cup 99	Hybrid machine learning method trained with random subsets of KDD Cup 99 dataset. Several ensemble approach usage	2016
Moustafa and Slay (2016)	125	N/A	KDD Cup 99 NSL-KDD UNSW-NB15	Comparison of databases in terms of complexity and usability for ML related techniques applied to IDSs	2016
Vasilomanolakis et al. (2015)	116	Neural Networks: HIDE method Several other Collaborative IDS architectures	N/A	Overview of Collaborative IDSs approaches and possible network related threats	2015
Ambusaidi et al. (2016)	111	Least square SVM	KDD Cup 99 NSL-KDD Kyoto2006 +	Supervised filter-based feature selection algorithm is presented for finding optimal data features for further classification	2016
Dhanabal and Shantharajah (2015)	97	Machine learning: J48 SVM Naïve Bayes	NSL-KDD	A thorough analysis of NSL-KDD database for IDS appliance. Additionally ML techniques are used in order to check NSL-KDD usability as a training data	2015
Weller-Fahy et al. (2015)	72	Machine Learning	N/A	Overview of multiple methods defining similarity and distance measures in area of Network Intrusion Anomaly Detection	2015
Iglesias and Zseby (2015)	61	Machine learning: Decision tree classifiers Naïve Bayes k-NN ANN SVM	NSL-KDD	Proposal of multi-stage feature selection method based on stepwise regression wrappers and filters	2015
David and Netanyahu (2015)	61	DBN	Custom malware database	Novel method of malware signature detection based on network traffic and host logs	2015
Ingre and Yadav (2015)	48	NN	NSL-KDD	Evaluation of NSL-KDD performance using NN related method for binary and five-class classification for each attack type	2015

type contains network distillation, adversarial training and classifier robustifying.

Defensive techniques for Neural Network adversarial attacks seem to be discussed in the literature very recently. This highlights the importance of the topics for contemporary Neural Network usage.

8 Conclusions

The paper summarizes the literature review performed in order to present neural network architectures usage for intrusion detection systems. We decided to perform it, as cyber security tends to be an emerging research topic and constant progression in the Neural network area is a fact. Neural network architectures are widely used for creating new models

for IDSs. Nevertheless, it is significant that they are quite often combined with other ML techniques in hybrid models, which show themselves as quite efficient solutions. Pure NN solutions may seem to be not sufficient to create highly operational solutions.

There is also a long-lasting challenge with available datasets for performing experiments for IDSs. There are several public datasets described in these articles, but all of them have their drawbacks, like age or record redundancy. They are also not always representative for real-life data. But on the other hand, due to be publicly available, they enable researchers to perform comparable benchmarking. Some experiments are being executed based on self-generated datasets. They may be more suitable for particular researchers groups need, but they are subject to privacy concerns.

One another important observation that came from out literature review, is the fact that NN are also quite often used for working on reduction of dimensionality of the data, generating signatures for datasets or general working on data preparation. High-dimensionality of data can cause inability of an effective training of ML algorithm and this is being currently spotted in the newest articles.

Quite significant is that nowadays researchers are looking for effective solutions for other fields then only "classic" computer network. Intrusion detection systems are now applied to the areas like internet of things, clouds, automotive, smart grids and mobile communication. Those all topics were covered by the articles we reviewed, simultaneously being emerging technological areas, where cybersecurity plays a crucial role. Therefore it is clearly shown that approaching challenges will be connected to the fact that new network protocols and network types are being created.

One of the biggest challenges we can see ahead in terms of intrusion detection systems is to have a possibility of creating system that could be reactive to any new and low frequent attacks. Currently available public databases are not a sufficient base for such a use case. One of the promising approaches that can be taken is focusing on particular types of attacks and preparing solution directly for them, as showed in couple of reviewed papers. This could make proposed solutions more adaptive to new types of threats. Additionally, what would have to be addressed is the enormous amount of data that are processed every day in the world. IDSs that will be created in future will have to be resistant to the problems connected with data volume.

It is worth to highlight that only basis of security implications of NN usage in IDSs are covered in this article. This is an important area of research that should not be neglected at the expense of studies on precision and performance of NN based IDSs.

Based on the conducted review we tried to create a coherent source of knowledge about NN appliance for IDSs. This work is meant as an overall introduction for future work in the field. We hope that all the solutions and datasets enlisted in this paper will enable researchers for more efficient and influential work regarding new IDS proposals.

References

Aburomman AA, Ibne Reaz MB (2016) A novel SVM-kNN-PSO ensemble method for intrusion detection system. Appl Soft Comput 38:360–372. https://doi.org/10.1016/j.asoc.2015.10.011

Agrawal S, Agrawal J (2015) Survey on anomaly detection using data mining techniques. Procedia Comput Sci 60:708–713. https://doi.org/10.1016/j.procs.2015.08.220

Ahmad I, Abdullah AB, Alghamdi AS (2009) Artificial neural network approaches to intrusion detection: a review. In: Proceedings of the 8th Wseas International Conference on Telecommunications and Informatics. World Scientific and Engineering Academy and Society (WSEAS), pp 200–205

Al-Yaseen WL, Othman ZA, Nazri MZA (2017) Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. Expert Syst Appl 67:296–303. https://doi.org/10.1016/j.eswa.2016.09.041

Alheeti KMA, Gruebler A, McDonald-Maier KD (2015) An intrusion detection system against malicious attacks on the communication network of driverless cars. In: 2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC). pp 916–921

Aljawarneh S, Aldwairi M, Yassein MB (2018) Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. J Comput Sci 25:152–160. https://doi.org/10.1016/j.jocs.2017.03.006

Almási A-D, Woźniak S, Cristea V et al (2016) Review of advances in neural networks: neural design technology stack. Neurocomputing 174:31–41. https://doi.org/10.1016/j.neucom.2015.02.092

Alom MdZ, Bontupalli V, Taha TM (2015) Intrusion detection using deep belief networks. In: 2015 National Aerospace and Electronics Conference (NAECON). pp 339–344

Ambusaidi MA, He X, Nanda P, Tan Z (2016) Building an intrusion detection system using a filter-based feature selection algorithm. IEEE Trans Comput 65:2986–2998. https://doi.org/10.1109/TC.2016.2519914

Amiri F, Rezaei Yousefi M, Lucas C et al (2011) Mutual information-based feature selection for intrusion detection systems. J Netw Comput Appl 34:1184–1199. https://doi.org/10.1016/j.jnca.2011.01.002

Ashfaq RAR, Wang X-Z, Huang JZ et al (2017) Fuzziness based semi-supervised learning approach for intrusion detection system. Inf Sci 378:484–497. https://doi.org/10.1016/j.ins.2016.04.019

Baracaldo N, Chen B, Ludwig H et al (2018) Detecting poisoning attacks on machine learning in IoT environments. In: 2018 IEEE International Congress on Internet of Things (ICIOT). pp 57–64

Besharati E, Naderan M, Namjoo E (2019) LR-HIDS: logistic regression host-based intrusion detection system for cloud environments. J Ambient Intell Humaniz Comput 10:3669–3692. https://doi.org/10.1007/s12652-018-1093-8

Borazjani PN, Everett CE, McCoy D (2014) OCTANE: An extensible open source car security testbed. In: Proceedings of the Embedded Security in Cars Conference. p 10

Bourlard H, Kamp Y (1988) Auto-association by multilayer perceptrons and singular value decomposition. Biol Cybern 59:291–294. https://doi.org/10.1007/BF00332918

Buczak AL, Guven E (2016) A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun Surv Tutor 18:1153–1176. https://doi.org/10.1109/COMST.2015.2494502

Camastra F, Ciaramella A, Staiano A (2013) Machine learning and soft computing for ICT security: an overview of current trends. J Ambient Intell Humaniz Comput 4:235–247. https://doi.org/10.1007/s12652-011-0073-z

Choo K-KR (2011) The cyber threat landscape: challenges and future research directions. Comput Secur 30:719–731. https://doi.org/10.1016/j.cose.2011.08.004

Corona I, Giacinto G, Roli F (2013) Adversarial attacks against intrusion detection systems: taxonomy, solutions and open issues. Inform Sci 239:201–225. https://doi.org/10.1016/j.ins.2013.03.022

Cunningham P, Delany SJ (2007) K-nearest neighbour classifiers. Mult Classif Syst 34:1–17

Darvish Rouani B, Samragh M, Javidi T, Koushanfar F (2019) Safe machine learning and defeating adversarial attacks. IEEE Secur Priv 17:31–38. https://doi.org/10.1109/MSEC.2018.2888779

David OE, Netanyahu NS (2015) DeepSign: deep learning for automatic malware signature generation and classification. In: 2015 International Joint Conference on Neural Networks (IJCNN). pp 1–8

Demetrio L, Biggio B, Lagorio G et al (2019) Explaining vulnerabilities of deep learning to adversarial malware binaries. https://arxiv.org/abs/1901.03583

Denning DE (1987) An intrusion-detection model. IEEE Trans Softw Eng SE 13:222–232. https://doi.org/10.1109/TSE.1987.232894

Dhanabal L, Shantharajah DSP (2015) A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. Int J Adv Res Comput Commun Eng 4:446–452

Du M, Li F, Zheng G, Srikumar V (2017) DeepLog: anomaly detection and diagnosis from system logs through deep learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. ACM, pp 1285–1298

Elman JL (1990) Finding structure in time. Cogn Sci 14:179–211. https://doi.org/10.1207/s15516709cog1402_1

Erfani SM, Rajasegarar S, Karunasekera S, Leckie C (2016) High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. Pattern Recognit 58:121–134. https://doi.org/10.1016/j.patcog.2016.03.028

Evgeniou T, Pontil M (2001) Support vector machines: theory and applications. In: Paliouras G, Karkaletsis V, Spyropoulos CD (eds) Machine learning and its applications: advanced lectures. Springer, Berlin, pp 249–257

Fadlullah ZMD, Tang F, Mao B et al (2017) State-of-the-art deep learning: evolving machine intelligence toward tomorrow's intelligent network traffic control systems. IEEE Commun Surv Tutor 19:2432–2455. https://doi.org/10.1109/COMST.2017.2707140

Guo Y, Liu Y, Oerlemans A et al (2016) Deep learning for visual understanding: a review. Neurocomputing 187:27–48. https://doi.org/10.1016/j.neucom.2015.09.116

Guresen E, Kayakutlu G (2011) Definition of artificial neural networks with comparison to other networks. Procedia Comput Sci 3:426–433. https://doi.org/10.1016/j.procs.2010.12.071

Harzing A-W (2007) Publish or Perish. In: Harzing.com. https://harzing.com/resources/publish-or-perish. Accessed 1 Apr 2019

Haykin S (1994) Neural networks: a comprehensive foundation, 1st edn. Prentice Hall PTR, USA

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9:1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hodo E, Bellekens X, Hamilton A et al (2016) Threat analysis of IoT networks using artificial neural network intrusion detection system. In: 2016 International Symposium on Networks, Computers and Communications (ISNCC). pp 1–6

De la Hoz E, De La Hoz E, Ortiz A et al (2015) PCA filtering and probabilistic SOM for network intrusion detection. Neurocomputing 164:71–81. https://doi.org/10.1016/j.neucom.2014.09.083

Huang G-B, Zhu Q-Y, Siew C-K (2006) Extreme learning machine: theory and applications. Neurocomputing 70:489–501. https://doi.org/10.1016/j.neucom.2005.12.126

Iglesias F, Zseby T (2015) Analysis of network traffic features for anomaly detection. Mach Learn 101:59–84. https://doi.org/10.1007/s10994-014-5473-9

Ingre B, Yadav A (2015) Performance analysis of NSL-KDD dataset using ANN. In: 2015 International Conference on Signal Processing and Communication Engineering Systems. pp 92–96

Javaid A, Niyaz Q, Sun W, Alam M (2016) A deep learning approach for network intrusion detection system. In: Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS). ICST, pp 21–26

KDD Cup (1999) KDD Cup 1999 Data. In: KDD Cup 1999 Data. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. Accessed 1 Jun 2019

Kang M-J, Kang J-W (2016) Intrusion detection system using deep neural network for In-vehicle network security. PLoS One. https://doi.org/10.1371/journal.pone.0155781

Kim J, Kim J, Thu HLT, Kim H (2016) Long short term memory recurrent neural network classifier for intrusion detection. In: 2016 International Conference on Platform Technology and Service (PlatCon). pp 1–5

Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 43:59–69. https://doi.org/10.1007/BF00337288

Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ (eds) Advances in neural information processing systems 25. Curran Associates, Inc., pp 1097–1105

Kwon D, Kim H, Kim J et al (2019) A survey of deep learning-based network anomaly detection. Clust Comput 22:949–961. https://doi.org/10.1007/s10586-017-1117-8

Kyoto dataset (2015) Traffic Data from Kyoto University's Honeypots. http://www.takakura.com/Kyoto_data/. Accessed 1 Jun 2019

LeNail A (2019) NN-SVG: publication-ready neural network architecture schematics. J Open Source Softw 4:747. https://doi.org/10.21105/joss.00747

Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86:2278–2324. https://doi.org/10.1109/5.726791

Liao H-J, Richard Lin C-H, Lin Y-C, Tung K-Y (2013) Intrusion detection system: a comprehensive review. J Netw Comput Appl 36:16–24. https://doi.org/10.1016/j.jnca.2012.09.004

Lin W-C, Ke S-W, Tsai C-F (2015) CANN: An intrusion detection system based on combining cluster centers and

nearest neighbors. Knowl-Based Syst 78:13–21. https://doi.org/10.1016/j.knosys.2015.01.009
Lincoln L (1998) DARPA 1998 & 1999 datasets. In: DARPA 1998 1999 Datasets. https://www.ll.mit.edu/r-d/datasets. Accessed 1 Apr 2020
Lippmann R, Haines JW, Fried DJ et al (2000a) The 1999 DARPA off-line intrusion detection evaluation. Comput Netw 34:579–595. https://doi.org/10.1016/S1389-1286(00)00139-0
Lippmann RP, Fried DJ, Graf I et al (2000b) Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation. In: Proceedings DARPA Information Survivability Conference and Exposition. DISCEX'00. pp 12–26 vol.2
Liu W, Wang Z, Liu X et al (2017) A survey of deep neural network architectures and their applications. Neurocomputing 234:11–26. https://doi.org/10.1016/j.neucom.2016.12.038
McHugh J (2000) Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. ACM Trans Inf Syst Secur TISSEC 3:262–294
Modi C, Patel D, Borisaniya B et al (2013) A survey of intrusion detection techniques in Cloud. J Netw Comput Appl 36:42–57. https://doi.org/10.1016/j.jnca.2012.05.003
Moustafa N, Slay J (2016) The evaluation of network anomaly detection systems: statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Inf Secur J Glob Perspect 25:18–31. https://doi.org/10.1080/19393555.2015.1125974
Moustafa N, Slay J (2015) 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). pp 1–6
NSL-KDD (2009) NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB. https://www.unb.ca/cic/datasets/nsl.html. Accessed 1 Jun 2019
Narudin FA, Feizollah A, Anuar NB, Gani A (2016) Evaluation of machine learning classifiers for mobile malware detection. Soft Comput 20:343–357. https://doi.org/10.1007/s00500-014-1511-6
Ozay M, Esnaola I, Yarman Vural FT et al (2016) Machine learning methods for attack detection in the smart grid. IEEE Trans Neural Netw Learn Syst 27:1773–1786. https://doi.org/10.1109/TNNLS.2015.2404803
Pandeeswari N, Kumar G (2016) Anomaly detection system in cloud environment using fuzzy clustering based ANN. Mob Netw Appl 21:494–505. https://doi.org/10.1007/s11036-015-0644-x
Pitropakis N, Panaousis E, Giannetsos T et al (2019) A taxonomy and survey of attacks against machine learning. Comput Sci Rev 34:100199. https://doi.org/10.1016/j.cosrev.2019.100199
Ranzato M, Poultney C, Chopra S, Cun YL (2007) Efficient learning of sparse representations with an energy-based model. In: Schölkopf B, Platt JC, Hoffman T (eds) Advances in veural information processing systems 19. MIT Press, pp 1137–1144
Rosenblatt F (1958) The perceptron: a probabilistic model for information storage and organization in the brain. Psychol Rev 65:386–408. https://doi.org/10.1037/h0042519
Saied A, Overill RE, Radzik T (2016) Detection of known and unknown DDoS attacks using artificial neural networks. Neurocomputing 172:385–393. https://doi.org/10.1016/j.neucom.2015.04.101
Shah B, Trivedi BH (2012) Artificial neural network based intrusion detection system: a survey. Int J Comput Appl 39:13–18
Shanmugavadivu R, Nagarajan N (2011) Network intrusion detection system using fuzzy logic. Indian J Comput Sci Eng IJCSE 2:101–111
Shone N, Ngoc TN, Phai VD, Shi Q (2018) A deep learning approach to network intrusion detection. IEEE Trans Emerg Top Comput Intell 2:41–50. https://doi.org/10.1109/TETCI.2017.2772792
Singh R, Kumar H, Singla RK (2015) An intrusion detection system using network traffic profiling and online sequential extreme learning machine. Expert Syst Appl 42:8609–8624. https://doi.org/10.1016/j.eswa.2015.07.015
Song J, Takakura H, Okabe Y et al (2011) Statistical analysis of honeypot data and building of Kyoto 2006 + dataset for NIDS evaluation. In: Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security. ACM, pp 29–36
Songma S, Chimphlee W, Maichalernnukul K, Sanguansat P (2012) Classification via k-means clustering and distance-based outlier detection. In: 2012 Tenth International Conference on ICT and Knowledge Engineering. pp 125–128
Stewart M (2019) Security vulnerabilities of neural networks. In: Medium. https://towardsdatascience.com/hacking-neural-networks-2b9f461ffe0b. Accessed 1 Jan 2020
Tang TA, Mhamdi L, McLernon D et al (2016) Deep learning approach for network intrusion detection in software defined networking. In: 2016 International Conference on Wireless Networks and Mobile Communications (WINCOM). pp 258–263
Tavallaee M, Bagheri E, Lu W, Ghorbani AA (2009) A detailed analysis of the KDD CUP 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications. pp 1–6
Tay A (2015) 6 common misconceptions when doing advanced Google Searching. http://musingsaboutlibrarianship.blogspot.com/2015/10/6-common-misconceptions-when-doing.html. Accessed 1 Apr 2019
Tran NN, Sarker R, Hu J (2018) An Approach for Host-Based Intrusion Detection System Design Using Convolutional Neural Network. In: Hu J, Khalil I, Tari Z, Wen S (eds) Mobile Networks and Management. Springer International Publishing, Cham, pp 116–126
UNSW-NB15 (2015) The UNSW-NB15 data set description. https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/. Accessed 1 Jun 2019
Vasilomanolakis E, Karuppayah S, Mühlhäuser M, Fischer M (2015) Taxonomy and survey of collaborative intrusion detection. ACM Comput Surv CSUR. https://doi.org/10.1145/2716260
Veen F van (2016) The Neural Network Zoo. In: Asimov Inst. https://www.asimovinstitute.org/neural-network-zoo/. Accessed 1 Jun 2019
Vinchurkar DP, Reshamwala A (2012) A review of intrusion detection system using neural network and machine learning technique. Int J Eng Sci Innov Technol IJESIT 1:10
Wang W, Zhu M, Zeng X et al (2017) Malware traffic classification using convolutional neural network for representation learning. In: 2017 International Conference on Information Networking (ICOIN). pp 712–717
Weller-Fahy DJ, Borghetti BJ, Sodemann AA (2015) A survey of distance and similarity measures used within network intrusion anomaly detection. IEEE Commun Surv Tutor 17:70–91. https://doi.org/10.1109/COMST.2014.2336610
Yin C, Zhu Y, Fei J, He X (2017) A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access 5:21954–21961. https://doi.org/10.1109/ACCESS.2017.2762418
Yuan X, He P, Zhu Q, Li X (2019) Adversarial examples: attacks and defenses for deep learning. IEEE Trans Neural Netw Learn Syst 30:2805–2824. https://doi.org/10.1109/TNNLS.2018.2886017
Zhang H, Cisse M, Dauphin YN, Lopez-Paz D (2018) Mixup: beyond empirical risk minimization. https://arxiv.org/abs/1710.09412