International Journal of Information & Network Security (IJINS)
Vol.3, No.3, June 2014, pp. 142~161
ISSN: 2089-3299

Machine Learning based Research for Network Intrusion Detection: A State-of-the-Art

Kanubhai K. Patel*, Bharat V. Buddhadev**
* CMPICA, Charotar University of Science & Technology
** Department of Computer Engineering, SS College of Engineering

Article Info

Article history:

Received May 5th, 2014
Revised May 20th, 2014
Accepted May 26th, 2014

Keyword:

Intrusion detection
Machine learning
Survey
Rule based learning
Misuse detection

ABSTRACT

This paper reviews the machine learning based research carried out for network intrusion detection to lead a secure computer and network systems to the extent possible. Starting with an initial set of about 460 research articles, more than 105 related studies in the period between 2000 and 2013 were selected focusing on using single or combine machine learning techniques for this review. Solutions using convergence of various techniques show a great promise and potential. Related studies are compared by their design, datasets used, and other experimental setups. Current achievements and limitations in developing intrusion detection systems using machine learning techniques are presented and discussed. We assume that reader has aware of basic concepts of machine learning techniques. A good number of future research directions are also provided.

Corresponding Author:

Kanubhai K. Patel,
CMPICA,
Charotar University of Science & Technology
Changa, India.
Email: kkpatel7@gmail.com

1. INTRODUCTION

Intrusion is a set of actions aimed to compromise integrity, confidentiality or availability (CIA) of computing resources in a computing environment [1]. In general terms, intrusive behaviour can be considered as any behaviour that deviates from normal use of the system. There are general four categories of intrusion [2] [3]:

— Denial of Service (DoS): The general task of DoS attacks is to interrupt some service on a host to prevent it from dealing with certain requests. For example, SYN flood, smurf and teardrop.

— Probing: It is to gain information about the target host. For example, port-scan and ping-sweep.

— User to Root (U2R): U2R attacks exploit vulnerabilities in operating systems and software to obtain root (administrator) access to the system. For example, buffer overflow attacks.

— Remote to Local (R2L): The intruder does not have an account on the host and attempts to obtain local access across a network connection. For example, password guessing attacks.

Intrusion Detection (ID) is the process of identifying and responding to intrusion activities [4]. ID is the process of monitoring and analyzing events that occur in a computer or network to detect behaviour of users that conflict with the intended use of the system [4]. An Intrusion Detection System (IDS) is software and/or hardware component that monitor the events in a computer or network, and analyze the activities for signs of possible violations of computer security policies. It employs techniques for modeling and recognizing intrusive behaviour in a computer system [4]. This is security system devoted to permanent inspection of computing environments, information technology (IT) infrastructure, and related assets such as hosts, networks, application, servers, and databases. The objective of IDS is to identify ongoing attacks in real time and to establish an active or passive response in order to prevent successful attacks. IDS is the 'burglar alarm' of the computer security field. IDS have been around since the 1980s. James Anderson [5] introduced the concept of host-based IDS in 1980. In 1987, Dorothy Denning presented IDS design [6]. Then, Heberlein et al. [7] introduced network-based IDS in 1990.

ID is an active research area in the field of computer and network security. Current ID technology often does not mature completely because of regular changes in computer, network and the general evolution of information and communication technology (ICT). Many improvements, from different perspectives, should be considered in ID technology, so that the challenging nature of the requirements for the current and future computing environments can be accommodated. Promising methodologies and technologies are required for the design and development of effective Intrusion Detection System (IDS). We have reviewed the research carried out for intrusion detection using machine learning techniques to lead a secure computer and network systems to the extent possible. Rule based learning is covered in greater detail and areas requiring further research and exploration are mentioned in the paper. Comparison among various machine learning techniques made by other researchers is also covered.

The rest of this article is organized as follows. Section-2 describes main two methods of intrusion detection viz. i) misuse detection, and ii) anomaly detection. While Section-3 describes main two approaches of intrusion detection viz. i) Stateful detection approach, and ii) Stateless detection approach. Section-4 briefly presents research in intrusion detection using various machine learning techniques. This section covers both (supervised and unsupervised) machine learning techniques for intrusion detection. Section-5 briefly describes comparison of various techniques made by researchers along with summarization. Section-6 briefly describes open research areas, future challenges and opportunities in the field of intrusion detection through machine learning techniques. Section-7 concludes the article.

2. INTRUSION DETECTION METHODS

There are two main detection methods, i) misuse detection, and ii) anomaly detection [8] [9] [4]. These terms are also known as knowledge based and behaviour based intrusion [10]. The misuse detection method attempts to encode knowledge of known intrusions (misuse or abuse), typically as rules, and use this to screen events (also known as a signature based IDS) [11] [12]. As per Gollman [9], misuse detection is successful in commercial IDS. Misuse detection method is although being effective against known attacks, it fails to protect from novel threats. This brings anomaly detection into the focus of security research (e.g., [13] [14] [15] [16]. The anomaly detection method attempts to 'learn' the features of event patterns that constitute normal behaviour, and, by observing patterns that deviate from established norms, detect when an intrusion has occurred [6].

In recent years, researchers have incorporated techniques that allow misuse detection systems to be more flexible, being capable of detecting more variations of attacks. This has been made possible with machine learning techniques such as artificial neural networks (ANN) and fuzzy logic, which are built to be able to generalize their models of known attacks to classify unseen cases. This is also the case for rule based systems, which were deemed in the past to be unable to detect even slight variations of attacks due to rigid rules [17]. Rule based systems are now also capable of detecting variations of attacks, and may even be employed for anomaly detection, largely due to researchers incorporating fuzzy logic to define the rules. A broad review of anomaly detection can be found in the work of [18] [19]. A review of the main techniques applied in data preprocessing for anomaly based network intrusion can be found in the work of Davis and Clark [20].

3. INTRUSION DETECTION APPROACHES

As per Engen [4], there are two main approaches to detect intrusions: i) Stateful, and ii) Stateless. Stateful approaches consider an attack as being composed of several events (stages), whilst stateless approaches attempt to classify single events as being an intrusion or not.

3.1. Stateful (Event correlation) detection approach

Engen [4] has considered event correlation synonymous with stateful approaches for simplicity, which has been subject to extensive research and is commonly adopted in commercial IDSs, e.g., HP OpenView [21], EMERALD eXpert and eXpert-BSM [22] [23], and Snort [24]. Event correlation systems may analyze data both spatially and temporally, building deterministic and/or probabilistic models of intrusions [25]. Spatial systems analyze events from different sources simultaneously, whilst temporal systems consider not only the order of events to be significant, but also the time between them. Rule based systems are commonly used for event correlation [25] [22] [4]. The system will filter events according to a set of rules (signatures) that determine the pattern of intrusions. Kruegel et al. [26] has proposed stateful intrusion detection for high-speed networks. Panichprecha et al. [27] presented an approach to multi-step scenario specification and matching. This aims to address some of the issues and problems inherent in to scenario specification and event correlation found in most previous research work.

3.2. Stateless intrusion detection approach

Stateless IDS is classifying single events (e.g., network connections) as being intrusive or normal. Stateless intrusion detection is popularly adopted in the data mining and machine learning communities, treating the intrusion detection problem as a classification task. We need to transform the raw data, such as tcpdump for network based IDS, into suitable feature vectors such as those in MADAM/ID [28]. The feature vectors may also include some a priori knowledge, such as the count feature in the KDD Cup'99 data set [29], which contains information about the number of connections from a particular user within the last two seconds. Here, a challenge is to obtain a feature (input variable) vector that is comprehensive enough to separate normal data from intrusive data, but also keep the size of this vector as small as possible. Normally, the problem is more difficult to solve if we have more features. For many machine learning algorithms, increasing the number of features (the dimension of the problem) significantly increases the training time required to learn the intrusion task. It also slows down the run-time and increase memory requirements with more features. This is commonly referred to as 'the curse of dimensionality' [30]. Hence, much research has been devoted to developing efficient techniques to perform feature selection [4]. A feature construction algorithm consists of two steps: feature extraction and feature selection [4].

4. MACHINE LEARNING TECHNIQUES FOR INTRUSION DETECTION

Machine Learning (ML) is a field of Artificial Intelligence (AI) that is concerned with constructing programs that can improve their behaviour with experience [31]. As per Mitchell [31],

> *"ML refers to algorithmic mechanisms that allow computers to learn from experience, examples and analogy."*

There are main two approaches of machine learning: i) Supervised learning, and ii) Unsupervised learning. In the case of intrusion detection, learning involves determining patterns of normal or intrusive behavior by examining the sample data. Within context of misuse detection, Sabhnani and Serpen [32] described application of machine learning techniques and algorithms to KDD Intrusion Detection Dataset, and also described why machine learning algorithms fail [33].

We have reviewed research in some of machine learning techniques viz. i) Rule based learning, ii) Decision tree, iii) Bayesian reasoning, iv) Neural Networks, v) Support Vector Machines (SVM), vi) Clustering, and vii) Nature-inspired. These techniques are presented in the next subsections.

4.1. Rule based learning

Many machine learning techniques are applied to the problem of intrusion detection, however, there are few that emphasize on automatic rule learning and a fewer that learn rules online (i.e., in a single-pass). Automatic rule learning for intrusion detection is an active area of research. Some of classifier algorithms of rules are JRIP, Decision Tabel, PART, and OneR. The prominent techniques for learning rules specifically for intrusion detection are: i) Rule based expert system, and ii) Fuzzy rule based.

4.1.1. Rule based expert system

Rule based expert system defines mechanisms to compare rules or signatures or scenarios against rule base or audit records. SRI International began research into an intrusion detection expert system in 1985 [34]. As a result of the research, the Intrusion Detection Expert System (IDES) has become a standard in intrusion detection systems. EMERALD [22], ASAX [35], and ORCHIDS [36] are the other examples of rule based expert system. In rule based expert system, the knowledge of human experts is encoded into a set of rules. EMERALD [22] is a forward chaining rule-based expert system. It generates a forward chain of rules which links audit records facts to signatures. While, ASAX [35] specifies signatures as pairs of conditions and actions. ORCHIDS [36] is based on the technique proposed in [37] whose idea is derived from ASAX. Here, the detection is performed by comparing events against application specific temporal logic expressions.

RIPPER (Repeated Incremental Pruning to Produce Error Reduction) [38] is a popular rule mining algorithm that can be used to create a classifier, which is considered in several studies, for e.g. [39] [28] [40]. RIPPER is a sequential covering based rule learner, extended from IREP (Incremental Reduced Error Pruning) [41], which has been used by several researchers for learning rules for intrusion detection. Apriori [42] learns association rules by mining the frequent episodes and has also been used for intrusion detection by good number of researchers. Ramesh and Mahesh [43] proposed a framework to learn rules in two stages. First, the sequential covering algorithm is used to learn highly accurate rules indicating the presence of a target class. In the second stage, rules classifying the negation of the target class are learnt on the subset covered collectively by all positive rules.

Mahoney and Chan [44] [45] introduced a randomized rule generation algorithm which they called LERAD (Learning Rules for Anomaly Detection). LERAD generates simple if then conditional rules similar to association rules. This system was extended also to learn rules from system call sequences [46]. Maloof [47] extended the AQ11 algorithm, the incremental version of the sequential covering based AQ algorithm to AQ11-PM (i.e., AQ11 with partial memory). JAM [48] and ADAM [49] mined association rules from the training data and then use them to detect intrusions in the test data. The JAM worked in a misuse detection mode while the ADAM in the anomaly detection mode. Vollmer, Alves-Foss, and Manic [50] presented a combined approach that uses GA and anomaly-based IDS to create rules for a signature-based IDS. They produced set of optimal rules (rule-based) for a specific, anomalous instance previously detected by an anomaly IDS. Srinivasa et al. [51] presented a rule based IDS in which they use genetic algorithm (GA) to make IDS more efficient. The advantages of the rule based expert system approach are i) the simplicity, and ii) straightforwardness of the signature matching mechanism. But, this technique will not perform well in a case where number of rules is large. Rule based expert system is the most suitable for misuse detection method. But it suffers from low flexibility and robustness.

4.1.2. Fuzzy rule based technique

Fuzzy logic [52] is an approach to obtain more flexible rules compared with crisp rule based expert systems. It is obvious that the nature of intrusion detection is fuzzy. Bridges and Vaughn [53] examined a combination of a rule based system (RBS) and fuzzy association rule mining to monitor network traffic and system audit trails. The RBS for misuse detection and the fuzzy association rule mining for anomaly detection. They found that fuzzy logic can help to extract more general patterns of intrusions. In their experiments, incorporating fuzzy logic into the rule mining reduced the number of false positives. Florez et al. [54] have extended the research of Bridges and Vaughn [53], making various improvements. Florez et al. [54] use prefix-trees for speeding the fuzzy association rule generation. In both studies above, a Genetic Algorithm (GA) has been used for feature selection and to optimize the fuzzy membership functions. Later, GAs have also been applied to rule learning by other researchers [55] [56] [57] and [58].

Dickerson et al. [59] developed the Fuzzy Intrusion Recognition Engine (FIRE) using fuzzy sets and fuzzy rules for detecting malicious activity in computer networks in a distributed fashion. FIRE is defined as a collection of autonomous agents. Each agent produces fuzzy information from input sensors. For each observed feature a fuzzy set is generated using a fuzzy c-means algorithm. Such information is combined by a fuzzy rule based system for determining the degree of normalcy. Experiments were conducted with synthetic data sets in three different scenarios: Port and Host scan, DoS attack and unauthorized services access. Tillapart et al. [60] proposed Fuzzy IDS (FIDS), a fuzzy rule based system. They provided numerous example rules.

Cho [61] used a fuzzy rule reasoning mechanism in order to detect an anomaly. The input of the fuzzy reasoning mechanism are Hidden Markov Model (HMM) evaluation values (from different HMM models). The fuzzy rules are designed according to the set of HMM’s. A centroid defuzzyfication technique was applied for determining the final classification (abnormal or normal). Experiments were conducted for detecting user-to-root attacks on data collected from graduate students. Su et al. [62] proposed a novel method of incremental mining. They implemented fuzzy association rules in a real-time NIDS. Owens and Levary [17] utilized fuzzy set theory to develop an adaptive expert system for network based intrusion detection. Fuzzy rules can be created to perform both misuse and anomaly detection. Events are mapped to fuzzy sets, which are then classified by an expert system that determines an alert with a suspicion level of either ‘low’, ‘medium’ or ‘high’. Jahromi and Taheri [63] proposed a method for learning rule weights in fuzzy rule-based classification systems. While Toosi and Kahani [64] proposed a novel approach based on an evolutionary soft computing model for intrusion detection using neuro-fuzzy classifiers.

Fuzzy association rule are employed by Tajbakhsh, Rahmati, and Mirzaei [65] for the building the classifier. Fries [66] proposed evolutionary optimization of a fuzzy rule based network IDS. It provides better performance in comparison to other evolutionary approaches. As per him, evolutionary based systems offer the ability to adapt to dynamic environments and thereby to identify unknown attack methods, while Fuzzy-based systems accommodate the fuzziness associated with altered and previously unidentified attack modes. Dhanalakshmi and Babu [67] proposed a system in which the fuzzy logic is integrated with the data mining methods using GA for intrusion detection. This system uses data mining to extract rules and Mamdami fuzzy inference system to determine the behaviour of the test data. Shanmugavadivu and Nagarajan [68] proposed anomaly based network IDS. They used fuzzy logic for identifying the intrusion activities in a network. This system generates fuzzy IF-THEN rules and with the help of fuzzy decision module the system identifies the appropriate classification of the test data. They used KDD Cup99 for the evaluation of IDS.

Om and Gupta [69] used fuzzy inference rules for host based IDS to monitor hardware profile changes and thereby to detect the unauthorized access in a computer system. A fuzzy logic technique has been used in correlation with ID because of its following characteristics [53] [59]:

- Various quantitative parameters used for Intrusion Detection e.g., CPU usage time, activity frequency, connection interval, etc., are fuzzy in nature [70].
- The concept of security itself is fuzzy as stated by Bridges et al. [70].
- Fuzzy systems can readily combine inputs from varying sources [13].
- The degree of alert that can occur with intrusion is often fuzzy [13].
- Fuzzy rules allow us to easily construct if-then rules that help in describing security attacks.

4.1.3. Association rule discovery

Association Rule mining is a very popular technique although it is very slow. It finds correlation between the attributes. It was initially applied to the so-called market basket analysis, which aims at finding regularities in shopping behavior of customers of supermarkets [71]. The concept of association rule mining for intrusion detection was introduced by Lee, et al. [72], and is extended by [73] [74] [75]. Disadvantages of Association rule discovery technique:

- The execution time or association rule approach increases exponentially with respect to time as the number of attributes increases [65].
- There is vast number of rules, it is not possible to process all rules in turn.

Audit Data Analysis and Mining (ADAM) [73] used classification algorithm and association rules to detect attacks in audit data. They try to improve the classification efficiency. Hanguang and Yu [76] applied the rule base deduced from Apriori algorithm to increase performance of the structure, which is the standard of the association rule mining.

4.1.4. Rule based languages

There are two important rule based languages, i) rule based sequence evaluation language (RUSSEL) [77], and ii) production-based expert system tool set (P-BEST) [22]. RUSSEL was used in the advanced security audit trail analysis on UNIX (ASAX) project [77]. It is flexible and better to describe sequential event patterns and corresponding actions. The disadvantage is that to specify an attack pattern, user needs to write a program. P-BEST was developed for the Multiplexed information and computing service (Multics) Intrusion Detection and Alerting System (MIDAS). It was employed by IDES, NIDES, and EMERALD [22] later. Advantages of P-BEST are:

- It has ability to invoke external C functions,
- It is a language pre-processor,
- It is quite small and intuitive, and
- It does not depend on the structure of the input data.

Disadvantages of P-BEST are:

- It is a low-level language.
- It is time consuming to specify the attack patterns.
- Correctness of the rules is difficult to check due to the interaction of the related rules.

4.2. Decision tree

Decision Tree (DT) is a simplest classifier. It uses a tree graph along with the probability to provide the best match for the input. DT is one of the most commonly used supervised learning techniques in IDS due to fast adaptation, its simplicity, and high detection accuracy. DT is widely used in misuse detection systems. It yields good performance and has benefits compare to other machine learning techniques. C4.5 algorithm [78] is the most popular DT classifier. NBTree [79] is a hybrid between decision tree and Naive Bayes. It creates trees whose leaves are NaiveBayes classifiers for the instances that reach the leaf. Other classifier algorithms based on DT are RandomForest and REPTree. As per Engen [4] drawbacks of DT are:

- It cannot generalize to new attacks in the same manner as certain other machine learning approaches (similar to that of rule based systems).
- It is not suitable for anomaly detection. Empirical findings also demonstrate that DT is very sensitive to the training data and does not learn well from imbalanced data [80].

Bouzida and Cuppens [81] applied DT for anomaly-based intrusion detection. They assign a default class to the test instance that is not covered by the tree. Then the default class is examined for unknown attack analysis. Peddabachigari et al. [82] presented an approach with DT and Support Vector Machines (SVM). They used the DARPA data set to DT and then passed through the SVM. An ensemble was then made from DT, SVM, and DT–SVM.

4.3. Bayesian reasoning

As per Mitchell [31]:

> *"Bayesian reasoning provides a probabilistic approach to inference. It is based on the assumption that the quantities of interest are governed by probability distributions and that optimal decisions can be made by reasoning about these probabilities together with observed data".*

There is a varied range of implementations of Bayesian reasoning for intrusion detection. Scott [83] used Bayesian reasoning for designing intrusion detection models. This includes modeling user commands, bursts of malicious network activity and network-level behavior.

There are two main Bayesian approaches, i) Bayesian Networks, and ii) Naive Bayes. Bayesian Networks was used in the decision process of hybrid system [84] [85]. Kruegel et al. [84] employed Bayesian Network to decide the final output classification. Mutz et al. [85] extended the work of Kruegel et al. [84], proposing an application based IDS that also considers system call arguments when analyzing user commands. ELICIT [86] used a Bayesian Network to process 16 terabytes of raw packets over 13 months to investigate insider threats. Bayesian networks are also used by [87] [88] for intrusion detection. According to Mitchell [31], there are two main disadvantages of Bayesian Networks, such as,

— The requirement of a priori knowledge about the problem to determine probabilities, and

— The method is computationally expensive.

Naive Bayes (NB) offers machine learning capabilities. It is a simplified version of Bayesian Networks. Mahoney and Chan [89] proposed Packet Header Anomaly Detection (PHAD) and Application Layer Anomaly Detection (ALAD). They give an excellent explanation of Bayes odds in packet headers. Newsome et al. [90] explained how to thwart conjunction and Bayes learners with two types of malicious training: a) red herring attacks to create false classifications; and b) inseparability attacks to blur distinctions between classes. Panda and Patra [91] proposed a NB based technique to detect signatures of specific attacks. Altwaijry and Algarny [92] proposed a multi-layer Bayesian based IDS. Multiple Bayesian filters in series with each filter optimized for a specific attack type achieves results that are better than what can be achieved by a single filter [92]. Advantages of Bayesian approach [93]:

— It handles situations with incomplete data sets,

— It allows one to learn about causal relationships,

— It is an ideal representation for combining prior knowledge, and

— It provides an efficient technique for avoiding the over fitting of data.

4.4. Neural Networks (NN)

Neural networks (NN) refers to the cluster of neurons that function or act together to solve a particular task and process information. These networks are also capable of learning through supervision or independently. NN can be classified into two types based on its architecture [94]: i) supervised neural network is the Multi-Level Perception (MLP), and ii) unsupervised neural networks is the Self-Organizing Maps (SOM). The popular NN technique for supervised learning is the Multi-Level Perceptron (MLP) and for unsupervised learning is Self-Organizing Maps (SOM) [95]. SOM can be useful for novelty detection, automated clustering and visual organization [95]. NN have been applied to solve the intrusion detection problem since the early 90s. Fox et al. [96] used a SOM for learning characteristics of normal activity, while Debar, Becker, and Siboni [97] have introduced NN for intrusion detection, as an alternative to statistical techniques, in the IDES (intrusion detection expert system).

4.4.1 Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN), as processing model, is inspired by the way nervous system work and they attempt to implement in computer systems neuron like capabilities. There are three layers in a typical ANN viz. i) input layer, ii) hidden layer, and iii) output layer. Each layer is composed of one or more nodes (neurons) and communication paths between them. All layers connected together form a network of nodes (or neurons). Normally information flows from the input to the output layer, although in some ANN architectures a feedback flow is present. The input layer represents the stimulus or information forwarded to the network, while the output layer is the final product of the neural processing. Input layer nodes often carry out hidden relationships amongst them producing “hidden” nodes. The hidden nodes and the interaction weight between input nodes compose the hidden layer.

Cannady [98] used a Cerebellar Model Articulation Controller (CMAC) neural network for detecting DoS attacks in a simulated environment. A CMAC neural network is a three layer feed-forward neural network that is capable of on-line learning. This property is used for Cannady to recognize new attacks: starting with a Ping-Flood attack and finishing with a UDP Packet Storm attack. Lippmann and Cunningham [99] developed a technique that combines keyword selection with neural networks. A feed-forward neural network is used for approximating posterior probabilities of telnet sessions with normal actions and with attacks. Experiments are conducted using the DARPA 98 data set.

Lee and Heinbuch [100] proposed a hierarchy of Back Propagation Neural Networks (BPNN) for detecting intrusions. Neural network in low levels are designed and trained with specific assertions about the network traffic. No network traffic data is used for training the neural networks. Neural networks in top-levels are used to combine the detection provided by low-level neural networks in order to increase the detection accuracy. Experiments were conducted with artificially generated data. Zhang and Wang [101] have also used BPNN approach for ID. BPNN has ability of accurate prediction and better persistence.

Han and Cho [102] described an evolutionary neural network based IDS, which has good detection performance and also reduces the training time. Om and Sarkar [103] proposed a neural network based model, which can detect changes in the hardware profile of a computer system. They used back propagation network for detection and reported that the very high and very low values of the learning rate have bad effect on the results. Choudhary and Swarup [104] proposed a neural network approach to improve the alert throughput of a network and making it attack prohibitive using IDS. The KDD CUP 99 dataset were used for evolving and testing intrusion. Using ANN, Herrero et al. [105] presented the MObile-Visualization Hybrid IDS (MOVIH-IDS) which provide visual results of a network sniffer on a mobile device.

4.4.2 Self-Organising Maps (SOM)

The SOM is an unsupervised learning model, i.e., it does not need labeled training data to build its model. This is a desirable feature for intrusion detection, since labeling thousands, or even millions, of records is a laborious task and mislabeling can occur. Furthermore, the SOM is more commonly applied to anomaly detection than any of the other ANN models. The SOM can be referred to as a clustering technique, which has the particular benefit of being capable of producing lower dimensional representations of multi-dimensional data, in what is referred to as a map. The SOM is trained on normal user data, which forms clusters of common behaviour of the user(s). Deviations from the main cluster(s) signify possible intrusions.

Rhodes, Mahaffey, and Cannady [106] explored the application of Kohonen SOM for characterizing the normal behavior of a computer network. The input of the Kohonen SOM was provided by a monitor stack that uses protocol analyzers for reducing and discriminating the network traffic. The approach was tested with buffer-overflow attacks. A SOM was applied to network based anomaly detection by [107], as a component of a hybrid IDS in addition to a DT and a RBS. The DT performs misuse detection and the RBS determines the final output based on individual outputs from the two former techniques. As per the findings of Pan [108], the DT does not detect the ftpwrite (R2L) attack, but the SOM does. However, with more false positives overall. Thames et al. [109] proposed a hierarchical hybrid IDS for network based intrusion detection, comprised of a SOM and NB. They adopted a SOM algorithm for supervised learning, so that it can learn and classify a proportion of intrusive data.

Gunes and Nur [110] proposed a visual SOM to build topological models of known attacks for forensic analysis. Langin et al. [111] used SOM in a model of detecting malignant network traffic. Mansour et al. [112] developed an IDS using unsupervised neural network, Kohonen maps. The method called as Performance-based Ranking Method was used on KDD data set. It works by deleting an input from the dataset and comparing the result before and after the deletion.

4.5. Support Vector Machines (SVM)

Support Vector Machines (SVM) is proposed by Vapnik in 1998 [113]. SVM classifier is designed for binary classification. It also provides a user specified parameter called penalty factor. It allows users to make a tradeoff between the number of misclassified samples and the width of a decision boundary. Eskin et al. [13] used SVM in addition to their clustering methods for unsupervised learning. Li et al. [114] introduced an IDS based on series of machine learning strategies, which has a following steps – compact data set is created by clustering the redundant data; apply the method ACO for selecting a proper small training data set; feature dimension are reduced from 41 to 19 so as to seize the key feature of the network; obtain the classifier with SVM and undertake a thorough prediction to the total KDD cup data set.

Chen and Chen [115] proposed a technique of intrusion detection using a hybrid SVM based on entropy and TF-IDF. While Horng et al. [116] suggested fusion of hierarchical clustering and SVM.

Hierarchical clustering provides the high qualified training instance to SVM reduces the training time and improve the performance of resultant SVM. Feature selection procedure was also applied to eliminate redundant features from the training set so that SVM model could classify the network data accurately. Overall performance was evaluated and is found to be worthy on comparing it with the other IDS.

4.6. Clustering

Clustering is unsupervised network intrusion detection technique. It finds patterns in unlabeled data with many dimensions (number of attributes). A good number of researchers have used clustering and outliers detection [117] [118] [119] for unsupervised network anomaly detection. The various clustering approaches are density-based methods, grid-based methods, model-based methods, partitioning methods, and hierarchy methods. Mostly k-Means clustering is used to find natural groupings of similar instances. Records that are far from any of these clusters indicate unusual activity that may be part of a new attack. Sequeira and Zaki [120] presented ADMIT, which is an anomaly-based IDS. It works by clustering its data to distinguish between normal and anomalous computer use. It can perform real-time intrusion detection and requires less training time than other methods. To reduce the data that the administrator must cope with, it only presents the centers of clusters to the administrator. In addition, their system had more correct categorization of behavior than another system using the same dataset.

Shah, Undercoffer, and Joshi [121] have used Principal Component Analysis (PCA) and fuzzy c-medoids clustering for generating a profile of normal activity in a computer system. PCA was used for reducing the dimensionality of the collected data and fuzzy c-medoids was used for generating the profiles representing normalcy. Experiments were conducted on data collected from UNIX and APACHE web servers. To reduce computational complexity, Leung and Leckie [118] proposed a grid based clustering algorithm. An unsupervised intrusion detection method was proposed by [122] by computing cluster radius threshold (CBUID). Siraj, Maarof, and Hashim [123] proposed an intelligent alert clustering model for network intrusion analysis. They used principal component analysis with expectation maximization technique to aggregate similar alerts and reduce the number of low quality alerts.

Advantages and disadvantages of clustering techniques are given by Chandola, Banerjee, and Kumar [18]. Bharti, Jain, and Shukla [124] proposed an intrusion detection model in which they used feature selection algorithm to select the non-redundant attributes. They used fuzzy K-mean clustering algorithm to calculate the membership of every data point and J48 classification techniques for assigning a cluster to a particular class. Lee et al. [125] employed K-means clustering with SOM so that the model developed becomes self-adaptive and dynamic in nature. Experiment were carried out on well-known data set KDD cup 99, and results shows that approach can growth the detection rate while making the false alarm rate low and also proficient of identifying new types of attacks.

Song et al. [126] reported an unsupervised auto-tuned clustering approach that optimizes parameters and detects changes based unsupervised anomaly detection for identifying unknown attacks. Casas et al. [127] proposed a novel unsupervised outlier detection technique based on combining subspace clustering and multiple evidence accumulation to detect intrusions. They evaluated the approach using two real time datasets and KDD cup 99.

4.7. Nature-inspired Techniques

Nature-inspired techniques are being applied to various domains of computer security including cryptology, secure protocol design and intrusion detection. Techniques like Artificial Immune Systems, Genetic Algorithms, Genetic Programming, and Swarm Intelligence are used for the detection of attack patterns, adaptively learning rules from network traffic and implementing the overall frameworks of intrusion detection and response systems [57]. In the next subsections we present a brief survey of some of the research in this area with a focus on the intrusion detection problem.

4.7.1 Artificial Immune System (AIS)

A good number of researchers are working on Artificial Immune System (AIS) in the last decade [128] [129]. The majority of applications of AIS to intrusion detection employ negative selection (NS) [128]. As this is more convenient to anomaly detection. Four main AIS models can be extracted from the literature [128] [130]: i) immune/idiotypic networks, ii) negative selection, iii) clonal selection, and iv) danger theory. Dasgupta and Gonzalez [129] developed a new version of the negative algorithm that uses real value representation. Elements of self/non-self-space are points in the n-dimensional euclidean space (Rn) while detectors are hyper-rectangles in Rn. Detectors were evolved using a genetic algorithm that maximizes the covering of the non-self-space while minimizing the matching self-points. A niching technique was used in order to evolve multiple detectors that cover cooperatively the entire non-self-space. Experiments were conducted on a subset of the DARPA99 data set and on a chaotic time series (Mackey-Glass). Gonzalez and

Dasgupta [131] combined AIS with a classification algorithm for anomaly detection. AIS is used for generating samples in the non-self-space and the classification algorithm is executed using the self (normal) and the artificial non-self (abnormal) samples. Each detector, generated with the AIS, is used for generating artificial abnormal samples - points inside the detector. Experiments are conducted on a subset of the DARPA99 data set and with a chaotic time series (Mackey-Glass). Kim et al. [128] provided a complimentary review of AIS applied to intrusion detection, while Timmis [130] provided reviews of general treatment of AIS.

4.7.2 Genetic Algorithm (GA)

Genetic Algorithm (GA) was originally introduced in the field of computational biology. It uses the computer to implement the natural selection and evolution. This concept comes from the ‘‘adaptive survival in natural organisms”. The algorithm starts by randomly generating a large population of candidate programs. Some type of fitness measure to evaluate the performance of each individual in a population is used. A large number of iterations is then performed that low performing programs are replaced by genetic recombination of high performing programs. That is, a program with a low fitness measure is deleted and does not survive for the next computer iteration. Researchers have tried to integrate it with IDS. Some researchers used GA for deriving classification rules [132] [133] [134]. GA is also used to optimize the features or parameters requirements of some core functions in which different AI techniques are used to derive rules [135] [53] [136].

Li [133] proposed a technique using GA to detect anomalous network intrusion [137] [138]. The technique considers quantitative and categorical features of network packet for deriving classification rules. Goyal and Kumar [139] proposed a GA based algorithm to classify attacks specifically all types of smurf attack. Wu and Banzhaf [140] proposed a computational multilevel selection framework. Wu and Banzhaf [141] presented a multilevel genetic programming approach based on their computational multilevel selection framework [140]. Xia et al. [142] used GA technique to detect anomalous network behaviors based on information theory [137] [138]. Gong, Zulkernine, and Abolmaesumi [137] presented an implementation of GA based approach to Network Intrusion Detection by deriving a set of classification rules. They proposed a support-confidence framework to judge fitness function. Abdullah et al. [138] proposed a GA based performance evaluation algorithm to network intrusion detection. They have used information theory to filter the traffic data. Mischiatti and Neri [143] used REGAL, a distributed GA for evolving rules, on intrusion detection data. The performance of REGAL is compared against RIPPER. Experiments were conducted on the Information Exploration Shootout (IES) contest. This data set contains logs collected at the gateway between a LAN and Internet.

Balajinath and Raghavan [144] have applied a GA to perform intrusion detection based on UNIX commands. They encode the commands with numeric values. Chittur [134] used a GA for evolving a decision tree with a randomized coefficient describing the data. The decision tree discriminates between normal and abnormal behavior. Experiments are conducted on the KDDCup 99 data set. Hossain and Bridges [145] proposed a framework for intrusion detection that combines fuzzy logic with genetic algorithms. First a set of fuzzy association rules extracted from audit data. Then, a genetic algorithm is used for tuning the fuzzy sets, describing the fuzzy association rules. Finally, the set of tuned fuzzy association rules are considered the profile of normal behavior in the computer system. No experiments are conducted.

Marin, Ragsdale, and Surdu [146] developed a hybrid technique that combines expert systems, clustering techniques, genetic algorithms, and Linear Vector Quantization (LVQ) for generating user profiles. The dimensionality of the data samples is reduced after they are clustered with a genetic algorithm. The LVQ refines the cluster centers once the dimensionality reduction process is done by the GA. Experiments are conducted on sequences of 5000 UNIX commands for a set of 50 users. Abadeh, Habibi, and Lucas [147] used fuzzy genetic based learning algorithm for intrusion detection. Lin and Wang [148] proposed a novel genetic clustering algorithm for intrusion detection. While Hoque, Mukit, and Bikas [149] applied GA to detect various types of network intrusions efficiently.

4.7.3 Swarm Intelligence

Swarm Intelligence can be categorized by growing behavior from swarming activity. Ant colony optimization (ACO), proposed by Colorni et al. [150], and particle swarm optimization (PSO), proposed by Eberhart and Kennedy [151] for intrusion detection are covered in this survey. Ramos and Abraham [152] have proposed ANTIDS (Ant IDS). They have compared ANTIDS with DT, SVM, and LGP. They described four advantages of ants:

— classification can be done in real time,

— new classes can be handled without retraining,

— learning can be either supervised or non-supervised and,

— the self-organizing nature makes it ideal for distributed IDS.

Tao et al. [153] proposed a fish swarm intelligence based algorithm for intrusion detection. Alam et al. [154] proposed a method for outlier detection that uses HPSO clustering based on swarm intelligence, which is capable of providing clustering at different levels of compactness. Wahid [155] proposed a novel Simplified Swarm Optimization (SSO) algorithm for i) a rule-based classifier and ii) feature selection.

As per Wahid [155],

> *"SSO is a simplified Particle Swarm Optimization (PSO) that has a self-organising ability to emerge in highly distributed control problem space, and is flexible, robust and cost effective to solve complex computing environments."*

Exchange Local Search (ELS) and Weighted Local Search (WLS), two local search strategies, proposed by Wahid [155] to improve SSO performance. Also a novel hybrid SSO-based Rough Set (SSORS) for feature selection has also been proposed for Network IDS. Amudha and Rauf [156] provided an overview of the research progress in swarm intelligence techniques to the problem of intrusion detection.

4.8. Summarized list of research

We have summarized research carried out for intrusion detection using machine learning techniques as shown in Table-1:

Table 1. Summarized List of Research for Intrusion Detection using Machine Learning Techniques

Sr.#	Machine learning Technique used	Researcher
1	Bayesian Reasoning	
	Bayesian Networks	[84] [85] [86] [87] [88]
	Naïve Bayes	[89] [90] [109] [91] [92]
2	Clustering	[117] [120] [121] [118] [123] [124] [119] [122] [126] [127] [125]
3	Decision Trees	[82] [81]
4	Nature-inspired	
	Artificial Immune System	[128] [130] [129] [131]
	Genetic Algorithms (GA)	[143] [144] [134] [145] [146] [148] [132] [133] [134] [147] [139] [142] [140] [141] [137] [138] [149]
	Swarm Intelligence	[153] [154] [155] [156]
5	Neural Networks	
	Artificial Neural Networks (ANN)	[97] [98] [99] [100] [105] [101] [104] [102] [103]
	Self-Organising Maps (SOM)	[96] [106] [107] [108] [109] [110] [111] [112]
6	Rule based learning	
	Rule based expert system	[34] [22] [40] [28] [48] [43] [49] [157] [46] [39] [50] [51]
	Fuzzy Rule based	[53] [59] [60] [17] [54] [61] [64] [67] [65] [62] [66] [68] [69]
	Association rule discovery	[72] [73] [74] [75]
7	Support Vector Machines (SVM)	[82] [13] [115] [114] [116]

5. COMPARISION OF MACHINE LEARNING TECHNIQUES

There is no single technique to detect all of the different attack types effectively. The IDS needs many diversified skill sets to handle many different attacks. A good number of researchers have performed direct comparisons between various machine learning techniques and combinations of techniques. In this section we described comparisons of machine learning techniques made by other researchers.

Mukkamala et al. [158] compared SVM with ANN and found several advantages to SVM. They have utilized host-based user activities for the comparisons. There is only one constrain with SVM, i.e. it provides binary classifications only. Web related data was also examined, such as the number of 404 errors. Lane and Brodley [159] compared HMM with Instance Based Learning (IBL). They reported that HMM yields better accuracy on detecting intrusions but there was no gain in classifying normal behaviour. Therefore, they put their results in the context of a hierarchical combination of classifiers, in which a 'light' classifier (such as IBL in this case) can serve as a filter of more obvious data (normal behaviour in this case) at a lower level, then at a higher level, a more comprehensive technique (such as HMM in this case) can be employed to classify the remaining data.

Amor, Benferhat, and Elouedi [79] compared the performance of NB and DT classifier using KDD 99 cup dataset. They found that NB classifier is 7 times faster than DT with respect to running time. Also, DT is comparatively better in classifying normal, DoS and R2L attacks. While NB classifier is better in classifying Probing and U2R attacks. Mukkamala et al. [161] used multivariate adaptive regression splines (MARS) and linear genetic programming (LGP) along with SVM and ANN. SVM is better than MARS and ANN in convergence and scalability, while LGP gave the overall best performance accuracy. Mukkamala et al. [160] compared resilient back propagation (RBP) and ANN. RBP beat other ANNs in terms of accuracy.

Chen et al. [166] also preferred SVM over ANN using the DARPA Data Set. Shah et al. [164] [165] compared an evolving fuzzy neural network (EFuNN) with ANN using Snort. Ourston et al. [162] compared Hidden Markov Models (HMM), DT and ANN based on network sensor alarms. They found HMM gave better performance. Peddabachigari, Abraham, and Thomas [163] compared DT and SVM. They found that DT is better than SVM in terms of overall accuracy. DT is much better than SVM, particularly in detecting U2R and R2L attacks. We have summarized comparison of various techniques of machine learning carried out by researchers in Table-2:

Table 2. Summarization of Comparison of Various Techniques made by Researchers

Techniques compared	Researcher	Remarks/Findings
SVM and ANN	[158]	SVM outperformed ANN
HMM and IBL	[159]	HMM yields better accuracy on detecting intrusions but there was no gain in classifying normal behaviour
DT and SOM	[108]	DT does not detect the *ftpwrite* (*R2L*) attack, but the SOM does
multivariate adaptive regression splines (MARS], linear genetic programming (LGP), ANN and SVM	[160]	SVM outperformed MARS and ANN in convergence and scalability, but LGP gave the overall best performance accuracy.
resilient back propagation (RBP), LGP, SVM, and ANNs	[161]	- RBP outperformed other ANNs. - LGP outperformed the SVMs and RBP in accuracy with the expense of time
HMM to DT and ANN	[162]	preferring HMM
DT and SVM	[163]	DT is much better in detecting U2R and R2L network attacks, compared to SVM.
Evolving fuzzy neural network (EFuNN) with typical ANN	[164] [165]	EFuNN was preferred
NB with DT	[79]	- NB is 7 times faster than DT - DT outperforms in classifying normal, denial of service (DoS), and remote to local (R2L) attacks, - NB classifier is superior in classifying Probing and user to root (U2R) attacks
SVM and ANN	[166]	SVM had fewer parameters to set than ANN
- hybrid FNT with PSO and - evolutionary algorithm with a hybrid ANN-PSO	[167]	Hybrid FNT outperformed the hybrid ANN in accuracy in most categories
ACO to DT, SVM, and LGP	[152]	Suggested advantages of ACO
Estimation of distribution algorithm (EDA) ANN (EDA-ANN) with PSO ANN and DT	[168]	EDA-ANN outperformed the others in accuracy in most categories followed by the PSO-ANN.
NB and BN	[169]	NB outperformed BN in identifying Internet Relay Chat (IRC) botnet traffic
LGP and Fuzzy rules classification	[170]	
three different kinds of genetic programming - Linear genetic programming (LGP), multi-expression programming (MEP), and gene expression programming (GEP) -	[171]	- MEP outperformed LGP in three attack classes, - LGP outperformed MEP in two attack classes. - GEP also obtained good results for all of the classes.
k-NN, fuzzy k-NN, evidence-theoretic k-NN, and fuzzy belief k-NN	[172]	fuzzy belief k-NN performed the best
LGP, SVM and MARS	[173]	LGP outperformed SVM and MARS
BN and DT (C4.5)	[174]	Mixed results on accuracy depending upon the attack type
Tabu Search (TS) cellular neural network (CNN), a genetic algorithm (GA) CNN, and a simulated annealing (SA) CNN with ANN	[175]	TS CNN performed best in terms of detection and false positive rates.
NB classifier with 5 other classifiers, i.e., JRip, Ridor, NNge, Decision Table, and Hybrid Decision Table	[176]	NB classifier is better than other 5 classifiers.
Support Vector Machine and Naïve Bayes Algorithms in Spam Classification	[177]	
DT (C4.5) and SVM	[178]	Accuracy and detection rate of C4.5 is higher than that of SVM, but false alarm rate of SVM is better.
SVM and Multi-Level Support Vector Machine (MLSVM)	[179]	MLSVM is more suitable than SVM.
analyse the performance of SOM to detect anomalies on KDD 99 and NSL-KDD datasets	[180]	Obtained 92.37% detection rate for KDD 99 dataset, while 75.49% detection rate for NSL-KDD dataset.

Chen et al. [166] [167] compared first a hybrid FNT with PSO, and secondly evolutionary algorithm with a hybrid ANN-PSO. Ramos and Abraham [152] compared ANTIDS with DT, SVM, and linear genetic programming (LGP). Four advantages of ACO were given: i) it provides online and real time classification; (ii) new classes can be handled without retraining; (iii) training can be either supervised or unsupervised; and, (iv) it supports distributed processing. Livadas et al. [169] compared Naive Bayes with Bayesian Networks in identifying Internet Relay Chat (IRC) botnet traffic [169]on wireless network traffic.

Chen et al. [168] compared an estimation of distribution algorithm (EDA) ANN with a PSO ANN, and a DT. The EDA-ANN is better than others in accuracy in most categories followed by the PSO-ANN. Abraham et al. [171] compared Linear genetic programming (LGP), multi-expression programming (MEP), and gene expression programming (GEP). Yang et al. [175] compared three types of cellular neural network (CNN): A Tabu Search (TS) CNN, a GA CNN, and a simulated annealing (SA) CNN. They found that TS CNN is the better than other in terms of detection and false positive rates.

Chou and Yen [172] compared four types of k-Nearest Neighbors (k-NN): k-NN, fuzzy k-NN, evidence-theoretic k-NN, and fuzzy belief k-NN. Fuzzy k-NN is comparatively the best in all. Mukkamala et al. [173] compared LGP with SVM and MARS. They found that LGP is better than SVM and MARS. It gives 100% detection rate for stealthy probes. Wang, Gombault, and Guyet [174] compared BN and DT (C4.5). Panda and Patra [176] compared NB with Ridor, JRip, , Hybrid Decision Table, Decision Table, and NNge. They found that NB is better than other classifiers. While, Wu and Yen [178] compared results of C4.5 and SVM. C4.5 is superior to SVM in accuracy and detection, but in false alarm rate, SVM is better.

McCue [177] compared four implementations of the Naive Bayes classification algorithm to those of the four different kernels available in an implementation of a Support Vector Machine (SVM) classifier by their accuracy, speed and effectiveness. Aghamohammadi and Analoui [179] compared SVM with Multi-Level SVM. They found that MLSVM is better than SVM. Ibrahim, Basheer, and Mahmod [180] analyzed the performance of SOM to detect anomalies on KDD 99 and NSL-KDD datasets of internet traffic activity simulation. They obtained 92.37% detection rate for KDD 99 dataset, while 75.49% detection rate for NSL-KDD dataset.

6. OPEN RESEARCH AREAS, FUTURE CHALLENGES AND OPPORTUNITIES

Intrusion Detection System (IDS) is a vital component of security measures shielding computer systems and networks from potential abuse and misuse. In 1980, John Anderson published one of the earliest papers on IDS in the Computer Security Threat Monitoring and Surveillance. Since then many different efficient approaches for IDS have been proposed and implemented in practice. However, the research on intrusion detection is still an active field and attracts attention of many researchers because of its challenges and necessity of IDS for our computing resources when using Internet. Some of challenges in current IDS are:

— Effectiveness: An IDS should detect attacks accurately without raising too many false alarms. It can be fine-tuned to produce less number of false alarms but only at the cost of increased number of false negatives (i.e., by missing the actual attacks); conversely, it can be made general to cover more attacks but only at the cost of increased number of false alarms. In addition, the efficiency of an intrusion detection system also contributes to determining its effectiveness.

— Adaptability: An IDS should perpetually learn changes in the environment over time and adjust to them accordingly. Adaptability is a major challenge and arguably the most desired characteristic for IDS. Generally, achieving adaptability automatically is a harder problem for misuse detection systems which rely on a manual creation of signatures. Anomaly detection systems by definition look for novel attacks but they also need to adapt their learnt models of normal behaviour relative to changes in the environment.

— Speed: dealing with high-performance network

— Diversity of environments: needs to operate in changing and adversarial network environments with diverse protocols, services, and applications.

— Fault tolerance: the IDS not becoming security vulnerability itself.

— Inter-Operability and transparency

— Ease of use.

— Timeliness: handle large amounts of data. Concerned with how quickly the IDS can propagate the information through the network to react to potential intrusions. It is also referred to as scalability.

Machine learning is becoming more and more important for solving these challenges as it gives computers the ability to learn without being explicitly programmed. However, one of the important research questions before machine learning can be applied for IDSs in practice is about the reliability of detection results provided by automatic learning algorithms. So far previous researches on intrusion detection have not studied this question well. Besides this, other challenge for machine learning is to obtain a feature (input variable or attribute) set that is comprehensive enough to separate normal data from intrusive data, but also keep the size of this set as small as possible. While the performance of machine learning in other areas is often determined by a single quality, such as the classification accuracy, security involves several factors that require attention.

Sommer and Paxson [181] have studied some of these factors for network intrusion detection and have summarized the main challenges that machine learning algorithms have to overcome in order to be useful in the field of intrusion detection. Rieck [182] extended this work to the generic application of machine learning and identify five key factors: effectively, efficiency, transparency, controllability, and robustness, that impact the efficacy of learning based security systems. Efficiency is often measured by the cost of learning and updating a detection model and the cost of actual detection in terms of both time and resources. Many conventional security instruments fail to satisfy all factors equally well [182]. For example, many tools for attack detection suffer from false alarms and analysis systems for malicious software are often vulnerable to evasion [182]. A substantial body of previous work on learning for security has ignored these factors and there is a clear demand for research that brings the promising capabilities of machine learning to practical security solutions [182]. There are several ways in which machine learning techniques can be applied to intrusion detection problems. For instance, as per Shafi [57], they can be used

— to automatically generate signatures or rules for misuse or signature based intrusion detection systems,

— in building and extracting interesting features that improve the effectiveness of existing detection systems, and

— to learn the normal behaviour of a protected system or its users in an anomaly detection context

7. CONCLUSION

In this paper, we review the research carried out along with tools and solutions available for intrusion detection to lead a secure computer and network systems to the extent possible. Solutions using convergence of various machine learning techniques show a great promise and potential. Still there is a good number of open challenges in the field to explore by researchers.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Ajay Parikh of Gujarat Vidyapith for providing help in this survey.

REFERENCES

[1] A Abraham and R Jain. (2004) Soft computing models for network intrusion detection systems. [Online]. http://arxiv.org/ftp/cs/papers/0405/0405046.pdf

[2] K Kendall, *A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems*.: Master’s thesis, Massachusetts Institute of Technology, 1999.

[3] R Lippmann et al., "Evaluating Intrusion Detection Systems: The 1998 DARPA Off-line Intrusion Detection Evaluation," in *DARPA Information Survivability Conference and Exposition*, vol. 2, 2000, pp. 12-26.

[4] V Engen, *Machine Learning For Network Based Intrusion Detection, PhD thesis, Bournemouth University, June 2010.*, 2010.

[5] James P Anderson, "Computer security threat monitoring and surveillance," Fort Washington, Pennsylvania, Technical report 1980.

[6] Dorothy E Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. 13, no. 2, pp. 222-232, February 1987.

[7] Todd L Heberlein et al., "A Network Security Monitor," in *IEEE Symposium on Security and Privacy*, 1990, pp. 296-305.

[8] S Axelsson, "Intrusion Detection Systems: A Taxonomy and Survey," Dept. of Computer Engineering, Chalmers University of Technology, Sweden, Technical Report 99-15, 2000.

[9] D Gollmann, *Computer Security*, 2nd ed.: Wiley, 2006.

[10] H Debar, M Dacier, and A Wespi, "Towards a Taxonomy of Intrusion Detection Systems," *Computer Networks*, pp. 805–822, 1999.

[11] v Paxson, "Bro: A System for Detecting Network Intruders in Real-Time," *Elsevier Computer Networks*, pp. 2435–2463, 1999.

[12] M Roesch, "Snort: Lightweight intrusion detection for networks," in *SENIX Large Installation System Administration Conference*, LISA, 1999, pp. 229–238.

[13] E Eskin, A Arnold, M Prerau, L Portnoy, and S Stolfo, "A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data," in *Applications of Data Mining in Computer Security*.: Kluwer, Dordrecht, 2002.

[14] C Kruegel, G Vigna, and W Robertson, "A multi-model approach to the detection of web based attacks," *Computer Networks*, vol. 48, no. 5, 2005.

[15] S J Stolfo et al., "A comparative evaluation of two algorithms for windows registry anomaly detection," *Journal of Computer Security*, pp. 659–693, 2005.

[16] R Perdisci, D Ariu, P Fogla, G Giacinto, and W Lee, "McPAD: A multiple classifier system for accurate payload-based anomaly detection," *Computer Networks*, vol. 5, no. 6, pp. 864–881, 2009.

[17] S F Owens and R R Levary, "An adaptive expert system approach for intrusion detection," *International Journal Secur. Netw.*, vol. 1, pp. 206–217, 2006.

[18] Varun Chandola, Arindam Banerjee, and Vipin Kumar, "Anomaly Detection: A Survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1-58, July 2009.

[19] C F Tsai, Y F Hsu, and C Y Lin, "Intrusion detection by machine learning: A review," *Expert Systems with Applications*, vol. 36, no. 10, pp. 11994-12000, 2009.

[20] Jonathan J Davis and Andrew J Clark, "Data preprocessing for anomaly based network intrusion detection: A review," *Computers & Security*, vol. 30, no. 6-7, pp. 353-375, June 2011.

[21] K Sheers, "HP OpenView Event Correlation Services," *Hewlett–Packard Journal*, vol. 11, pp. 31–42, 1996.

[22] U Lindqvist and P A Porras, "Detecting Computer and Network Misuse Through the Production-Based Expert System Toolset (P-BEST)," in *IEEE Symposium on Security and Privacy*, 1999, pp. 146–161.

[23] U Lindqvist and P A Porras, "eXpert-BSM: A Host Based Intrusion Detection Solution for Sun Solaris," in *17th Annual computer Security Applications Conference*, New Orleans, USA, 2001, pp. 240–251.

[24] Sourcefire Inc. (1999) Sourcefire Inc Snort.

[25] Guofei Jiang and George Cybenko, "Functional Validation in Grid Computing," *Autonomous Agents and Multi-Agent Systems*, vol. 8, no. 2, pp. 119-130, 2004.

[26] Christopher Kruegel, Fredrik Valeur, Giovanni Vigna, and Richard Kemmerer, "Stateful Intrusion Detection for High-Speed Networks," in *2002 IEEE Symposium on Security and Privacy*, Berkeley, CA, 2002, pp. 285-294.

[27] Sorot Panichprecha, Jacob Zimmermann, George Mohay, and Andrew Clark, "Multi-Step Scenario Matching Based on Unification," in *5th Australian Digital Forensics Conference*. Perth, Western Australia, 2007, pp. 88-97.

[28] W Lee and S J Stolfo, "A Framework for Constructing Features and Models for Intrusion Detection Systems," *ACM Transactions on Information and System Security*, vol. 3, pp. 227–261, 2000.

[29] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[30] R O Duda, P E Hart, and D G Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.

[31] Mitchell, *Machine learning*. New york: McGraw Hill, 1997.

[32] Maheshkumar Sabhnani and Gursel Serpen, "Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context," in *International Conference on Machine Learning, Models, Technologies and Applications (MLMTA 2003)*, vol. 1, Las Vegas, NV, USA, 2003, pp. 209-215.

[33] Maheshkumar Sabhnani and Gursel Serpen, "Why Machine Learning Algorithms Fail in Misuse Detection on KDD Intrusion Detection Data Set," *Journal of Intelligent Data Analysis*, 2004.

[34] T F Lunt, "Real-Time Intrusion Detection," in *IEEE COMPCON*, 1989.

[35] Abdelaziz Mounji, "Languages and Tools for Rule-Based Distributed Intrusion Detection," University of Namur, Belgium, PhD thesis 1997.

[36] Julien Olivain and Jean Goubault-Larrecq, "The ORCHIDS Intrusion Detection Tool," in *17th International Conference on Computer Aided Verification (CAV'05)*, vol. volume 3576 of Lecture Notes in Computer Science, Edinburgh, Scotland, UK, 2005.

[37] Muriel Roger and Jean Goubault-Larrecq, "Log Auditing Through Model Checking," in *14th IEEE Computer Security Foundations Workshop (CSFW'01)*, Cape Breton, Nova Scotia, Canada, 2001, pp. 220–236.

[38] W W Cohen, "Fast Effective Rule Induction," in *12th International Conference on Machine Learning*, 1995, pp. 115–123.

[39] J He, D Long, and C Chen, "An Improved Ant-based Classifier for Intrusion Detection," in *Third International Conference on Natural Computation (ICNC 2007)*, Washington, DC, USA, 2007, pp. 819–823.

[40] C Warrender, S Forrest, and B Pearlmutter, "Detecting Intrusions Using System Calls: Alternative Data Models," in *IEEE Symposium on Security and Privacy*, p. 1999.

[41] J Furnkranz and G Widmer, "Incremental reduced error pruning," in *Eleventh International Conference on Machine Learning*, 1994, pp. 70-77.

[42] R Agrawal and R Srikant, "Fast algorithms for mining association rules," in *20th International Conference on Very Large Data Bases*, 1994, pp. 487-499.

[43] A Ramesh and J V Mahesh, "PNrule: A new framework for learning classifier models in data mining (a case-study in network intrusion detection)," in *First SIAM International Conference on Data Mining*, Chicago, IL USA, 2001.

[44] M Mahoney, "A Machine Learning Approach to Detecting Attacks by Identifying Anomalies in Network Traffic," Florida Institute of Technology, Ph. D. Thesis 2003.

[45] M V Mahoney and P K Chan, "Learning rules for anomaly detection of hostile network traffic," in *Third IEEE*

International Conference on Data Mining (ICDM 2003), 2003, pp. 601-604.

[46] G Tandon and P Chan, "Learning Useful System Call Attributes for Anomaly Detection," in *18th International FLAIRS Conference*, 2005, pp. 405-411.

[47] M Maloof, "Incremental rule learning with partial instance memory for changing concepts," in *International Joint Conference on Neural Networks*, vol. 4, 2003, pp. 2764-2769.

[48] S J Stolfo, W Fan, W Lee, A Prodromidis, and P K Chan, "Cost-based Modeling and Evaluation for Data Mining With Application to Fraud and Intrusion Detection: Results from the JAM Project," in *DARPA Information Survivability Conference*, 2000, pp. 130-144.

[49] D Barbara, N Wu, and S Jajodia, "Detecting Novel Network Intrusions Using Bayes Estimators," in *First SIAM Conference on Data Mining*, 2001.

[50] T Vollmer, J Alves-Foss, and M Manic, "Autonomous rule creation for intrusion detection," in *2011 IEEE Symposium on Computational Intelligence in Cyber Security*, 2011, http://www.inl.gov/technicalpublications/Documents/5025964.pdf.

[51] K G Srinivasa, S Chandra, S Kajaria, and S Mukherjee, "IGIDS: Intelligent Intrusion Detection System Using Genetic Algorithms," in *World Congress on Information and Communication Technologies*, 2011, pp. 852-857.

[52] L A Zadeh, "Fuzzy sets.," *Information and Control*, vol. 8, pp. 338–353, 1965.

[53] S M Bridges and R B Vaughn, "Fuzzy Data Mining And Genetic Algorithms Applied To Intrusion Detection," in *12th Annual Canadian Information Technology Security Symposium*, 2000, pp. 109-122.

[54] G Florez, S M Bridges, and R B Vaughn, "An Improved Algorithm for Fuzzy Data Mining for Intrusion Detection," in *North American Fuzzy Information Processing Society Conference NAFIPS-FLINTS 2002*, 2002, pp. 457–462.

[55] S Jeya and K Ramar, "Rule Based Network Intrusion Detection System Based on Crossover and Mutation," *Asian Journal of Information Technology*, vol. 6, pp. 896–901, 2007.

[56] S Selvakani and R S Rajesh, "Genetic algorithm for framing rules for intrusion detection," *International Journal of Computer Science and Network Security*, pp. 285–290, 2007.

[57] K Shafi, T Kovacs, H A Abbass, and W Zhu, "Intrusion detection with evolutionary learning classifier systems," *Natural Computing: an international journal*, vol. 8, pp. 3–27, 2009.

[58] A Orfila, J M Estevez-Tapiador, and A Ribagorda, "Evolving High-Speed, Easy-to-Understand Network Intrusion Detection Rules with Genetic Programming," in *EvoWorkshops '09: Proc. of the Evo-Workshops 2009 on Applications of Evolutionary Computing*, Berlin, Heidelberg, 2009, pp. 93-98.

[59] J E Dickerson, J Juslin, O Koukousoula, and J A Dickerson, "Fuzzy intrusion detection," in *North American Fuzzy Information Processing Society Conference (NAFIPS-FLINTS 2001)*, 2001.

[60] P Tillapart, T Thumthawatworn, and P Santiprabhob, "Fuzzy intrusion detection system," *Assumption University Journal of Technology (AU J.T.)*, vol. 6, no. 2, pp. 109–114, 2002.

[61] S Cho, "Incorporating soft computing techniques into a probabilistic intrusion detection system," *IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews*, vol. 32, no. 2, pp. 154-160, 2002.

[62] M-Y Su, G-J Yu, and C-Y Lin, "A real-time network intrusion detection system for large-scale attacks based on an incremental mining approach," *Comput Security*, vol. 75, pp. 301–309, 2009.

[63] M Zolghadri Jahromi and M Taheri, "A proposed method for learning rule weights in fuzzy rule-based classification systems," *Fuzzy Sets and Systems*, vol. 159, pp. 449–459, 2007.

[64] A N Toosi and M Kahani, "A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers," *Computer Communications*, vol. 30, pp. 2201–2212, 2007.

[65] Arman Tajbakhsh, Mohammad Rahmati, and Abdolreza Mirzaei, "Intrusion detection using fuzzy association rules," *Applied Soft Computing*, vol. 9, pp. 462–469, 2009.

[66] Terrence P Fries, "Evolutionary optimization of a fuzzy rule-based network intrusion detection system," Dept. of Computer Science, Indiana University, 2010 Annual Meeting 2010.

[67] Y Dhanalakshmi and Ramesh I Babu, "Intrusion detection using data mining along fuzzy logic and genetic algorithms," *International Journal of Computer Science and Network Security*, vol. 8, no. 2, pp. 27–32, 2008.

[68] R Shanmugavadivu and N Nagarajan, "An Anomaly Based Netwok Intrusion Detection System Using Fuzzy logic," *International Journal of Computer Science and Information Security*, vol. 8, no. 8, pp. 185-193, 2010.

[69] Hari Om and Alok Kumar Gupta, "Design of Host based Intrusion Detection System using Fuzzy Inference Rule," *International Journal of Computer Applications (0975 – 8887)*, vol. 64, no. 9, February 2013.

[70] M Crosbie and G Stafford, "Applying genetic programming to intrusion detection," in *AAAI Symposium on Genetic Programming*, Cambridge, MA., 1995, pp. 1-8.

[71] Margaret H Dunham, *Data Mining Introductory and Advanced Topics*.: Prentice Hall, 2003.

[72] W Lee and S J Stolfo, "Data Mining Approaches for Intrusion Detection," in *7th USENIX Security Symposium*,

1998, pp. 79-94.

[73] D Barbara, J Couto, S Jajodia, and N Wu, "ADAM: A Testbed for Exploring the Use of Data Mining in Intrusion Detection," *SIGMOD Record*, vol. 30, no. 4, pp. 15-24, 2001.

[74] W Lee, S Stolfo, and M Kui, "A Data Mining Framework for Building Intrusion Detection Models," in *IEEE Symposium on Security and Privacy*, 1999, pp. 120-132.

[75] S Manganaris, M Christensen, D Zerkle, and K Hermiz, "A Data Mining Analysis of RTID Alarms," in *Recent Advances in Intrusion Detection, Second International Workshop*, 1999.

[76] Li Hanguang and Ni Yu, "Intrusion Detection Technology Research Based on Apriori Algorithm," *Physics Procedia*, vol. 24, pp. 1615–1620, 2012.

[77] A Mounji, B L Charlier, D Zampuniéris, and N Habra, "Distributed audit trail analysis," in *ISOC'95 symposium on network and distributed system security*, Los Alamitos, CA, 1995, pp. 102--112.

[78] J Quinlan, *C4.5: Programs for Machine Learning*. San Mateo: Morgan Kaufmann, 1993.

[79] N B Amor, S Benferhat, and Z Elouedi, "Naïve Bayes vs. Decision Trees in Intrusion Detection Systems," in *2004, ACM Symposium on Applied Computing*, 2004, pp. 420-424.

[80] F Gharibiana and A A Ghorbani, "Comparative Study of Supervised Machine Learning Techniques for Intrusion Detection," in *Fifth Annual Conference on Communication Networks and Services Research (CNSR '07)*, Washington, DC, USA, 2007, pp. 350–358.

[81] Y Bouzida and F Cuppens, "Detecting known and novel network intrusions," in *Security and Privacy in Dynamic Environments, IFIP International Federation for Information Processing Volume 201.*, 2006, pp. 258-270.

[82] S Peddabachigari, A Abraham, C Grosan, and J Thomas, "Modeling intrusion detection system using hybrid intelligent systems.," *Journal Network Computer Application*, vol. 30, no. 1, pp. 114–132, 2007.

[83] S L Scott, "A bayesian paradigm for designing intrusion detection systems," *Computational Statistics Data Analysis*, vol. 45, no. 1, pp. 69–83, 2004.

[84] C Kruegel, D Mutz, W Robertson, and F Valeur, "Bayesian Event Classification for Intrusion Detection," in *19th Annual Computer Security Applications Conference (ACSAC)*, 2003.

[85] D Mutz, F Valeur, G Vigna, and C Kruegel, "Anomalous system call detection," *ACM Transaction on Information System Securrity*, vol. 9, pp. 61–93, 2006.

[86] M A Maloof and G D Stephens, "Elicit: a system for detecting insiders who violate need-to-know," in *10th international symposium on Recent advances in intrusion detection (RAID 2007)*, vol. volume 4637 of Lecture Notes in Computer Science, Gold Coast, Australia, 2007, pp. 146-166.

[87] M Mehdi, A Zair, A Anou, and M Bensebti, "A Bayesian Networks in Intrusion Detection Systems," *Journal of Computer Science*, vol. 3, no. 5, pp. 259-265, 2007.

[88] A Cemerlic, L Yang, and J Kizza, "Network Intrusion Detection Based on Bayesian Networks," in *Twentieth International Conference on Software Engineering and Knowledge Engineering (SEKE'2008)*, San Francisco, CA, USA, 2008.

[89] M V Mahoney and P K Chan, "Learning nonstationary models of normal network traffic for detecting novel attacks," in *8th ACM SIGKDD international conference on knowledge discovery and data mining*, 2002, pp. 376–385.

[90] J Newsome, B Karp, and D Song, "Paragraph: thwarting signature learning by training maliciously," in *Recent advances in intrusion detection, 9th international symposium (RAID 2006)*, vol. 4219 of Lecture Notes in Computer Science, Hamburg, Germany, 2006, pp. 81-105.

[91] M Panda and M R Patra, "Network intrusion detection using naïve Bayes," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 7, no. 12, pp. 258-263, December 2007.

[92] Hesham Altwaijry and Saeed Algarny, "Multi-Layer Bayesian Based Intrusion Detection System," in *World Congress on Engineering and Computer Science 2011 Vol II (WCECS 2011)*, vol. II, San Francisco, USA, 2011.

[93] David Heckerman, "A Tutorial on Learning with Bayesian Networks," 2006.

[94] Shelly Xiaonan Wu and Wolfgang Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," *Applied Soft Computing*, vol. 10, no. 1, pp. 1-35, January 2010.

[95] V K Pachghare, Parag Kulkarni, and Deven M Nikam, "Intrusion Detection System Using Self Organizing Maps," , 2009.

[96] K Fox, R Henning, J Reed, and R Simonian, "A neural network approach towards intrusion detection," in *13th National Computer Security Conference*, 1990.

[97] H Debar, M Becker, and D Siboni, "A Neural Network Component for an Intrusion Detection System," in *IEEE Computer Society Symposium on Research in Security and Privacy*, Los Alamitos Oakland, CA, 1992, pp. 240–250.

[98] J Cannady, "Next generation intrusion detection: Autonomous reinforcement learning of network attacks," in *Twenty Third National Information Security Conference*, 2000, pp. 1-12.

[99] R P Lippmann and R K Cunningham, "Improving intrusion detection performance using keyword selection and

neural networks," *Computer Networks*, vol. 34, no. 4, pp. 597–603, 2000.

[100] S Lee and D Heinbuch, "Training a neural-network based intrusion detector to recognize novel attacks," *IEEE Transactions on Systems, Man and Cybernetics, Part A (Systems and Humans)*, vol. 31, no. 4, pp. 294–299, July 2001.

[101] Wei Zhang and Hao-yu Wang, "Intrusive Detection Systems Design based on BP Neural Network," in *IEEE*, 2010.

[102] S J Han and S B Cho, "Evolutionary Neural Network for Anomaly Detection Based on the Behaviour of a Program," *IEEE Trans. on Systems, Man and Cybernetics-Part B*, vol. 36, no. 3, pp. 559-570, 2006.

[103] H Om and T K Sarkar, "Neural network based intrusion detection system for detecting changes in hardware profile," *Journal of Discrete Mathematics and Cryptography*, vol. 12, no. 4, pp. 451-466, 2009.

[104] Amit Kumar Choudhary and Akhilesh Swarup, "Neural Network Approach for Intrusion Detection," in *ICIS 2009*, Seoul, Korea, 2009.

[105] A Herrero, E Corchado, M A Pellicer, and A Abraham, "Movih-ids: a mobile-visualization hybrid intrusion detection system," *Neuro computing*, vol. 72, pp. 2775–2784, 2009.

[106] B Rhodes, J Mahaffey, and J Cannady, "Multiple self-organizing maps for intrusion detection," in *Twenty Third National Information Security Conference*, 2000.

[107] O Depren, M Topallar, E Anarim, and M K Ciliz, "An intelligent intrusion detection system (ids) for anomaly and misuse detection in computer networks," *Expert System with Applications*, vol. 29, no. 4, pp. 713–722, 2005.

[108] Z S Pan, S C Chen, G B Hu, and D Q Zhang, "Hybrid Neural Network and C4.5 for Misuse Detection," *Machine Learning and Cybernetics*, pp. 2463–2467, 2003.

[109] J L Thames, R Abler, and A Saad, "Hybrid intelligent systems for network security," in *44th annual Southeast regional conference*, New York, NY, USA, 2006, pp. 286–289.

[110] Kayacik H Gunes and Zincir-Heywood A Nur, "Using self organizing maps to build an attack map for forensic analysis," in *ACM international conference on privacy, security, and trust (PST 2006)*, 2006, pp. 285–293.

[111] C Langin, H Zhou, B Gupta, S Rahimi, and M R Sayeh, "A self organizing map and its modeling for discovering malignant network traffic," in *2009 IEEE symposium on computational intelligence in Cyber Security*, Nashville, TN, USA, 2009.

[112] Mansour M Alsulaiman, Aasem N Alyahya, Raed A Alkharboush, and Naseer S Alghafis, "Intrusion Detection System using Self-Organizing Maps," in *Third International Conference on Network and System Security, 978-0-7695-3838-9/09*, 2009.

[113] Vapnik, *Statistical learning theory.* New York: John Wiley, 1998.

[114] Y Li et al., "An efficient intrusion detection system based on support vector machines and gradually feature removal method," *Expert Systems with Applications*, vol. 39, pp. 424-430, 2011.

[115] R C Chen and S P Chen, "Intrusion Detection Using a Hybrid Support Vector Machine Based on Entropy and TF-IDF," *International Journal of Innovative Computing, Information and Control (IJICIC)*, vol. 4, no. 2, pp. 413-424, 2008.

[116] S Horng et al., "A novel intrusion detection system based on hierarchical clustering and support vector machines," *Expert Systems with Applications*, vol. 38, pp. 306–313, 2011.

[117] L Portnoy, E Eskin, and S Stolfo, "Intrusion detection with unlabeled data using clustering," , 2001.

[118] K Leung and C Leckie, "Unsupervised anomaly detection in network intrusion detection using clusters," in *Twenty-eighth Australasian conference on Computer Science*, vol. 38, Darlinghurst, Australia, 2005, pp. 333-342.

[119] M H Bhuyan, D K Bhattacharyya, and J K Kalita, "NADO: network anomaly detection using outlier approach," in *International Conference on Communication, Computing & Security*, New York, NY, USA, 2011, pp. 531-536.

[120] K Sequeira and M Zaki, "Admit: anomaly-based data mining for intrusions," in *KDD '02: Eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 2002, pp. 386–395.

[121] H Shah, J Undercoffer, and A Joshi, "Fuzzy clustering for intrusion detection," in *Third International Conference FUZZ-IEEE*, 2003.

[122] S Jiang, X Song, H Wang, J -J Han, and Q -H Li, "A clustering-based method for unsupervised intrusion detections," *Pattern Recognition Letter*, vol. 27, no. 7, pp. 802–810, May 2006.

[123] M M Siraj, M A Maarof, and S Z.M. Hashim, "Intelligent Alert Clustering Model for Network Intrusion Analysis," *International Journal of Advance Soft Computer Applications*, vol. 1, no. 1, pp. 33-48, 2009.

[124] K Bharti, S Jain, and S Shukla, "Fuzzy K-mean Clustering Via J48 For Intrusion Detection System," *International Journal of Computer Science and Information Technologies*, vol. 1, no. 4, pp. 315-318, 2010.

[125] S Lee, G Kim, and S Kim, "Self-adaptive and dynamic clustering for online anomaly detection," *Expert Systems with Applications*, vol. 38, pp. 14891–14898, 2011.

[126] J Song, H Takakura, Y Okabe, and K Nakao, "Toward a more practical unsupervised anomaly detection system," *Information Sciences*, August 2011.

[127] P Casas, J Mazel, and P Owezarski, "Unsupervised network intrusion detection systems: Detecting the unknown

without knowledge," *Computer Communications*, January 2012.

[128] J Kim et al., "Immune system approaches to intrusion detection – a review," *Natural computing*, vol. 4, pp. 413–466, December 2007.

[129] D Dasgupta and F Gonzalez, "An immunity-based technique to characterize intrusions in computer networks," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 3, pp. 281–291, June 2002.

[130] J Timmis, "Artificial immune systems – today and tomorrow," *Natural Computing*, vol. 6, pp. 1–18, 2007.

[131] F Gonzalez and D Dasgupta, "Neuro-immune and self-organising map approaches to anomaly detection: A comparison," in *First International Conference on Artificial Immune Systems*, 2002.

[132] M M Pillai, J. H. P Eloff, and H S Venter, "An Approach to Implement a Network Intrusion Detection System using Genetic Algorithms," in *SAICSIT'04: the 2004 annual research conference of the South African institute of computer scientists and and information technologists on IT research in developing countries South African Institute for Computer Scientists and Information Technologists*, Republic of South Africa, 2004, pp. 221-221.

[133] W Li, "A Genetic Algorithm Approach to Network Intrusion Detection," Technical report 2004.

[134] A Chittur. (2005) Model Generation for an Intrusion Detection System Using Genetic Algorithms. [Online]. http://ids.cs.columbia.edu/sites/default/files/gaids-thesis01.pdf

[135] Jonatan Gomez and Dipankar Dasgupta, "Evolving Fuzzy Classifiers for Intrusion Detection," in *the 2002 IEEE Workshop on Information Assurance*, 2002.

[136] M Middlemiss and G Dick, "Feature selection of intrusion detection data using a hybrid genetic algorithm/KNN approach," *Design and application of hybrid intelligent systems*, pp. 519-527, 2003.

[137] Ren Hui Gong, Mohammad Zulkernine, and Purang Abolmaesumi, "A Software Implementation of a Genetic Algorithm Based Approach to Network Intrusion Detection," in *Sixth International Conference on Software Engineering, Artificial Intelligence,Networking and Parallel/Distributed Computing & 1st ACIS Int. Workshop on Self-Assembling Wireless Networks*, 2005.

[138] B. Abdullah, I Abd-alghafar, Gouda I Salama, and A Abd-alhafez, "Performance Evaluation of a Genetic Algorithm Based Approach to Network Intrusion Detection System," in *13th International Conference on AEROSPACE SCIENCES & AVIATION TECHNOLOGY (ASAT-13)*, Kobry Elkobbah, Cairo, Egypt , 2009, pp. 1-17.

[139] Anup Goyal and Chetan Kumar. (2008) GA-NIDS: A Genetic Algorithm based Network Intrusion Detection System. [Online]. http://www.cs.northwestern.edu/~ago210/ganids/GANIDS.pdf

[140] Shelly Xiaonan Wu and Wolfgang Banzhaf, "A hierarchical cooperative evolutionary algorithm," in *12th Genetic and Evolutionary Computation Conference (GECCO '10)*, Portland, OR, USA, 2010, pp. 233-240.

[141] Shelly Xiaonan Wu and Wolfgang Banzhaf, "Rethinking Multilevel Selection in Genetic Programming," in *Conference of Genetic and Evolutionary Computation (GECCO'11), July 12–16, 2011*, Dublin, Ireland, 2011.

[142] T Xia, G Qu, S Hariri, and M Yousif, "An efficient network intrusion detection method based on information theory and genetic algorithm," in *the 24th IEEE International Conference on Performance, Computing and Communications*, Phoenix, Arizona, USA, 2005, pp. 11-17.

[143] M Mischiatti and F Neri, "Applying local search and genetic evolution in concept learning systems to detect intrusion in computer networks," in *Workshop about Machine Learning and Data Mining. Seventh conference AI*IA "Intelligenza Artificiale"*, 2000.

[144] B Balajinath and S V Raghavan, "Intrusion Detection Through Learning Behaviour Model," *International Journal of Computer Communications*, vol. 24, pp. 1202–1212, July 2001.

[145] M Hossain and S M Bridges, "A framework for an adaptive intrusion detection system with data mining," in *13th Annual Canadian Information Technology Security Symposium*, 2001.

[146] J Marin, D Ragsdale, and J Sirdu, "A hybrid approach to the profile creation and intrusion detection," in *DARPA Information Survivability Conference and Exposition II. DISCEX'01*, vol. 1, 2001.

[147] M S Abadeh, J Habibi, and C Lucas, "Intrusion Detection Using a Fuzzy Genetic-Based Learning Algorithm," *Journal of Network and Computer Application*, vol. 30, pp. 414-428, 2005.

[148] C C Lin and M S Wang, "Genetic-clustering algorithm for intrusion detection system," *International Journal of Information Computer Security*, vol. 2, no. 2, pp. 218-234, 2008.

[149] Mohammad Sazzadul Hoque, Md. Abdul Mukit, and Abu Naser Md. Bikas, "An Implementation of Intrusion Detection System using Genetic Algorithm," *International Journal of Network Security & Its Applications (IJNSA)*, vol. 4, no. 2, pp. 109-120, March 2012.

[150] A Colorni, M Dorigo, and V Maniezzo, "Distributed optimization by ant colonies," in *European conference on artificial life*, Paris, France, 1991, pp. 134–142.

[151] R Eberhart and J Kennedy, "A new optimizer using particle swarm theory," in *Sixth international symposium on micro machine and human science*, 1995.

[152] V Ramos and A Abraham, "Antids: self organized ant-based clustering model for intrusion detection system," in *Fourth IEEE international workshop on soft computing as transdisciplinary science and technology (WSTST'05)*, 2005, pp. 977–986.

[153] L Tao, H Yuan-bin, Q Ai-ling, and C Xin-Tan, "Feature optimization based on artificial fish-swarm algorithm in intrusion detection," in *2009 international conference on networks, security, wireless communications and trusted computing*, Hube, Wuhan, 2009, pp. 542-545.

[154] S Alam, G Dobbie, P Riddle, and M A Naeem, "A swarm intelligence based clustering approach for outlier detection," 2010.

[155] Noorhaniza Wahid, "A Novel Approach to Data Mining Using Simplified Swarm Optimization," The University of Sydney, PhD Thesis 2011.

[156] P Amudha and Abdul H Rauf, "A Study on Swarm Intelligence Techniques in Intrusion Detection," *IJCA*, no. Computational Intelligence & Information Security, pp. 9-16, November 2012.

[157] G. Helmer, J Wong, V Honavar, and L Miller, "Automated discovery of concise predictive rules for intrusion detection," *The Journal of Systems and Software*, no. 60, pp. 165-175, 2002.

[158] S Mukkamala, G Janoski, and A Sung, "Monitoring systsem security using neural networks and support vector machines," in *International workshop on hybrid intelligent systems*, 2001, pp. 121–138.

[159] T Lane and C E Brodley, "An Empirical Study of Two Approaches to Sequence Learning for Anomaly Detection," *Machine Learning*, vol. 51, pp. 73–107, 2003.

[160] S Mukkamala, A Sung, and A Abraham, "Designing intrusion detection systems: architectures and perspectives," in *The international engineering consortium (IEC) annual review of communications*, vol. 57, 2004, pp. 1229–1241.

[161] S Mukkamala, A H Sung, and A Abraham, "Modeling intrusion detection systems using linear genetic programming approach," in *17th international conference on industrial and engineering applications of artificial intelligence and expert systems*, vol. 3029 of Lecture Notes in Computer Science, 2004, pp. 633-642.

[162] D Ourston, S Matzner, W Stump, and B Hopkins, "Coordinated internet attacks: responding to attack complexity," *Journal of Computer Security*, vol. 12, pp. 165–190, 2004.

[163] S Peddabachigari, A Abraham, and J Thomas, "Intrusion detection systems using decision tress and support vector machines," *International Journal of Applied Science and Computations*, 2004.

[164] K Shah et al., "Adaptive neuro-fuzzy intrusion detection system," in *IEEE international conference on ITCC'04*, vol. 1, 2004, pp. 70–74.

[165] S Chavan, K Shah, N Dave, and S Mukherjee, "Adaptive neuro-fuzzy intrusion detection systems," in *IEEE international conference on information technology: coding and computing (ITCC'04)*, Los Alamitos, CA, 2004, pp. 70-74.

[166] W-H Chen, S-H Hsu, and H-P Shen, "Application of SVM and ANN for Intrusion Detection," *Comput Operation Research*, vol. 32, no. 10, pp. 2617–2634, 2005.

[167] Y Chen, A Abraham, and J Yang, "Feature deduction and intrusion detection using flexible neural trees," in *Second IEEE International Symposium on Neural Networks (ISNN 2005)*, 2005.

[168] Y Chen, Y Zhang, and A Abraham, "Estimation of distribution algorithm for optimization of neural networks for intrusion detection system," in *Artificial intelligence and soft computing—ICAISC 2006*, 2006.

[169] C Livadas, B Walsh, D Lapsley, and T Strayer, "Using machine learning techniques to identify botnet traffic," in *Second IEEE LCN workshop on network security (WNS)*, Tampa, FL, USA, 2006.

[170] A Abraham, R Jain, J Thomas, and SY Han, "D-scids: distributed soft computing intrusion detection system," *Journal of Network Computer Application*, vol. 30, pp. 81–98, 2007.

[171] A Abraham, C Grosan, and C Martin-Vide, "Evolutionary design of intrusion detection programs," *International Journal of Network Security*, vol. 4, no. 3, pp. 328–339, 2007.

[172] T-S Chou and KK Yen, "Fuzzy belief k-nearest neighbors anomaly detection of user to root and remote to local attacks," in *2007 IEEE workshop on information assurance, United States Military Academy*, West Point, NY, 2007, pp. 207–213.

[173] S Mukkamala, A Sung, and A Abraham, "Hybrid multi-agent framework for detection of stealthy probes.," *Appl Soft Computing Journal*, vol. 7, no. 3, pp. 631–641, 2007.

[174] W Wang, S Gombault, and T Guyet, "Towards fast detecting intrusions: using key attributes of network traffic," in *Third international conference on internet monitoring and protection*, 2008, pp. 86–91.

[175] Z Yang, A Karahoca, N Yang, and N Aydin, "Network intrusion detection by using cellular neural network with tabu search," in *Bio-inspired learning and intelligent systems for security, 2008 (BLISS'08)*, 2008.

[176] M Panda and M R Patra, "Semi-naïve Bayesian method for network intrusion detection system," in *16th International Conference on Neural Information Processing*, 2009.

[177] Rita McCue. (2009, November) University of California at Santa Cruz. [Online]. http://classes.soe.ucsc.edu/cmps242/Fall09/proj/RitaMcCueReport.pdf

[178] S Y Wu and E Yen, "Data mining-based intrusion detectors," *Expert Systems with Applications*, vol. 36, pp. 5605-5612, 2009.

[179] Milad Aghamohammadi and Morteza Analoui, "A Comparison of Support Vector Machine and Multi-Level

Support Vector Machine on Intrusion Detection," in *World of Computer Science and Information Technology (WCSIT)*, vol. 2 (7), 2012, pp. 215-219.

[180] LAHEEB M IBRAHIM, DUJAN T BASHEER, and MAHMOD S MAHMOD, "A Comparison Study For Intrusion Database (Kdd99, NSL-KDD) Based On Self Organization Map (SOM) Artificial Neural Network," *Journal of Engineering Science and Technology*, vol. 8, no. 1, 2013.

[181] Robin Sommer and Vern Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in *2010 IEEE Symposium on Security and Privacy (SP '10)*, Oakland, California, 2010, pp. 305-316.

[182] Konrad Rieck, "Computer Security and Machine Learning: Worst Enemies or Best Friends?," Technische University, Berlin, Germany, PhD Thesis 2011.