Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 07, No. 04, December 2021.

NETWORK INTRUSION DETECTION SYSTEMS USING SUPERVISED MACHINE LEARNING CLASSIFICATION AND DIMENSIONALITY REDUCTION TECHNIQUES: A SYSTEMATIC REVIEW

Zein Ashi, Laila Aburashed, Mahmoud Al-Qudah and Abdallah Qusef

(Received: 21-Aug.-2021, Revised: 4-Nov.-2021, Accepted: 14-Nov.-2021)

ABSTRACT

Protecting the confidentiality, integrity and availability of cyberspace and network (NW) assets has become an increasing concern. The rapid increase in the Internet size and the presence of new computing systems (like Cloud) are creating great incentives for intruders. Therefore, security engineers have to develop new technologies to match growing threats to NWs. New and advanced technologies have emerged to create more efficient intrusion detection systems using machine learning (ML) and dimensionality reduction techniques, to help security engineers bolster more effective NW Intrusion Detection Systems (NIDSs). This systematic review provides a comprehensive review of the most recent NIDS using the supervised ML classification and dimensionality reduction techniques, it shows how the used ML classifiers, dimensionality reduction techniques and evaluating metrics have improved NIDS construction. The key point of this study is to provide up-to-date knowledge for new interested researchers.

KEYWORDS

Network intrusion detection, Machine learning, Supervised learning, Dimensionality, Systematic review.

1. INTRODUCTION

With the new development in NWs and communications, cybersecurity has become a vital requirement to defend new cyber-attacks [1]. Recently, IDSs in general and NIDSs in particular, have been increasingly used as tools to constantly monitor NW traffic and provide desired security protection against cyber-attacks [2]. The earliest IDS was produced in 1980 by Jim Anderson and since then, such systems have continuously developed and improved, to keep pace with the rapid growth in the NW and communication fields [3]. The growth of cyberspace has introduced the Big Data concept to the IDS field, in which massive volumes of data are continually generated around the Internet. Security engineers have used this Big Data with different ML techniques for further IDS improvements [1]. Supervised ML NIDS depends on pre-collected datasets to learn how to distinguish between normal and abnormal NW traffic, to be able to detect any intrusions in the future [3].

The main purpose of this systematic review is to provide a broad analysis of developments in modern supervised ML NIDSs. The core idea is to provide updated information on supervised ML NIDSs to provide a starting point for new researchers who want to explore this field. This study undertakes three main objectives to contribute to existing knowledge: (1) To conduct a systematic review of selected research papers concerned with supervised ML NIDS published during 2017 and until March 2021 in Science-Direct (Elsevier), Springer-Link (Springer) and IEEE-Explore (IEEE) libraries; (2) To review each research paper extensively and discuss its used ML classifiers, dimensionality reduction algorithms and evaluation metrics; and (3) To highlight recent trends in using such technologies for building NIDSs and various future challenges.

There are many survey papers in the literature providing reviews on NIDSs, but this study is unique in applying a systematic approach to collect more relevant research papers on NIDSs designed by supervised ML classification and dimensional minimization techniques. This study reviews the most recent research papers from the past three years, providing up-to-date knowledge for researchers.

Section 2 reviews related studies in this area to present background information about IDSs and Section 3 details IDS categorization. Section 4 explains the research methodology, followed by the application

Z. Ashi, L. Aburashed, M. Al-Qudah and Abdallah Qusef (Corresponding Author) (ORCID: 0000-0003-4769-6992) are with the King Hussein School of Computing Science, Princess Sumaya University for Technology, Amman, Jordan. Emails: zai20198121@std.psut.edu.jo, lay20188108@std.psut.edu.jo, mah20208173@std.psut.edu.jo and a.qusef@psut.edu.jo

of supervised ML and dimensionality reduction techniques in Section 5. Section 6 presents the evaluation metrics. Section 7 discusses the salient findings and identified challenges, while Section 8 concludes the paper.

2. RELATED WORKS

Numerous researchers have taken an interest in NIDSs and machine learning and a variety of surveys and systematic reviews have summarized previous studies in this field. Zebari *et al.* [29] conducted a comprehensive review of dimensionality reduction techniques used in the previous IDSs. For each study, they provided some details about the algorithms used, datasets, dimensionality reduction techniques (categorized into feature selection and feature extraction) and they summarized the achieved results. Although they analyzed recent studies (between 2018 and 2020), they did not follow a systematic approach, unlike the current study (which provides a systematic approach to collect the analyzed research papers, to make the data collection more accurate and comprehensive).

Martins *et al.* [56] presented a systematic review of ML-based systems to detect intrusion and malware scenarios. They reviewed 20 research papers from multiple scientific e-libraries and compared them based on attack techniques, used algorithms, datasets, evaluation metrics and their results. The limitation of their study was that they did not provide details about their systematic approach and did not mention whether the analyzed studies were recent or not. In our study, we provide a detailed description for our followed approach.

Ahmad *et al.* [1] reviewed recent studies (from 2017 to 2020) that generally used machine learning and deep-learning techniques. Their review was notable in identifying the strengths and weaknesses for each reviewed study, which we have also applied in the current work.

Some studies that introduced the software system IDS were analyzed by Ramaki *et al.* [57]. They limited their study to ML techniques that used “Hidden Markov” models and did not provide a systematic approach for collecting research papers for analysis. This systematic review spans a wider domain, including NIDSs based on supervised ML techniques. Gonzalez *et al.* [58] developed a method for improving security inside secure military self-protected software and comprehensively analyzed software present position and potential responses to threats. Their method consisted of three stages: user detection, analysis of current situation and reactive action. The detection phase consists of analyzing location, timing at present location and identifying user type (friend or foe). The analysis phase entails determining whether self-protected software should be present at the current site, predicting future locations and analyzing the level of hazard at the current location. Analytical results showed that self-protected software that incorporates user detection provides higher protection than self-protected software that does not contain such detection capability.

Nassif *et al.* [59] analyzed ML approaches utilized to detect cloud system attacks with a detailed systematic review for 63 relevant research papers from 2004 to 2021. For each study, they identified the related security threats, ML techniques used and evaluation metrics’ results. This systematic review provides more comparison criteria between the analyzed research papers.

3. IDS CATEGORIZATION

An IDS system identifies abnormal events by constantly monitoring network traffic, keeping a network log and alerting the network administrator in the event of any intrusion. IDS copies the network traffic for read-only analysis, to detect any suspicious events and notify the administrators about what is going on (to take manual responsive actions). IDS is implemented outbound of the network line, without affecting the network data flow [4]. IDS can be categorized based on its monitoring environment and detection approach. Types of IDS according to monitoring environment are host-based (HIDS), NW-based (NIDS) and hybrid IDS, while according to their detection approaches they can be classified as signature-based, anomaly-based and hybrid [5] (Figure 1).

3.1 IDSs According to Monitoring Environment

HIDS operates in a local machine that detects local abnormal behaviours; any changes to the host registry, unauthorized access attempts or attacks cannot be detected by firewalls [5]. IDS is considered

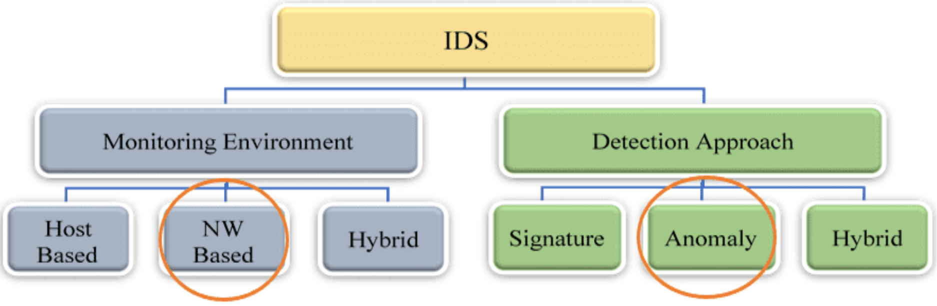


Figure 1. IDS categories and types.

a reliable system, because it analyzes the log files so that it can efficiently determine whether an attack is active or not [6]. NIDS operates in an NW node used to monitor and analyze network traffic on a single network node to detect any abnormal traffic [7]. Some NIDSs are created by analyzing the payload of an NW packet (packet level) or analyzing only that packet's header (flow level) [7]. Hybrid IDS integrates HIDS and NIDS in an effective way [2].

3.2 IDSs According to Detection Approach

Signature-based IDS (also called "misuse detection IDS" or "knowledge-based IDS") uses a blacklist of predefined intrusions and attacks. When any intrusion in the blacklist occurs, this IDS can detects it accurately, with no false alarms [8]. The disadvantages of this type are the required storage size and that it cannot detect any novel predefined intrusion on its blacklist. This blacklist requires constant updates to be able to detect any new intrusions [2].

Anomaly-based IDS (also called "behaviour-based IDS") uses the definition of the normal NW traffic and any deviation of that normality is detected as an intrusion. It compares the actual NW traffic with the predefined characteristics of normal traffic to detect any intrusions [9]. It can detect any novel intrusion, but it suffers high false alarms, as it is difficult to define uniform traffic among all NWs [2].

Hybrid IDS efficiently combines signature-based and anomaly-based approaches, to detect known attacks in the blacklist while simultaneously detecting new ones [2]. Anomaly-based NIDS is the main focus of this study, developed using supervised, unsupervised or reinforcing ML techniques [7].

4. Research Methodology

The methodology used in this study is adapted from [10], collecting papers on NIDSs built with ML techniques published from 2017 to March 2021 in the Science-Direct (Elsevier), Springer-Link (Springer) and IEEE-Explore (IEEE) libraries. Search keywords have been used to achieve results related to the search questions, according to the inclusion and exclusion criteria phases (Figure 2).

4.1 Research Questions

RQ1. What are the proposed supervised ML classification and dimensionality reduction techniques used to build the NIDS?

This question describes the supervised ML classification and dimensionality reduction techniques which have been used in previous studies to build NIDS against cyber-attacks, to investigate the popular techniques used for more enhancement in this domain.

RQ2. What are the evaluation metrics used to evaluate the proposed NIDS?

RQ3. What are the best supervised ML classification and dimensionality reduction techniques used to build the NIDS?

The main purpose of NIDSs is to detect intrusions in real time, with high sensitivity and low false alarms. This question explores whether the built NIDS provides a noticeable enhancement in this domain, as well as to identify techniques that enhance NIDS sensitivity without increasing processing overhead or affecting real time detection.

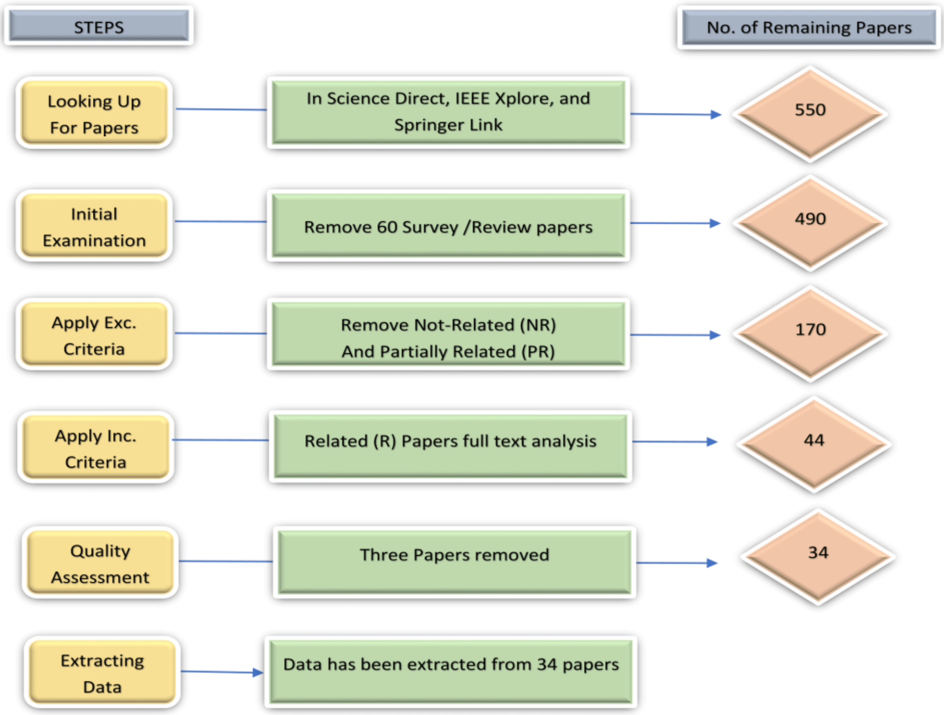

STEPS
No. of Remaining Papers
Looking Up For Papers
In Science Direct, IEEE Xplore, and Springer Link
550
Initial Examination
Remove 60 Survey /Review papers
490
Apply Exc. Criteria
Remove Not-Related (NR) And Partially Related (PR)
170
Apply Inc. Criteria
Related (R) Papers full text analysis
44
Quality Assessment
Three Papers removed
34
Extracting Data
Data has been extracted from 34 papers

Figure 2. Flow process of inclusion and exclusion of papers.

4.2 Research Process

4.2.1 E-Library Search Phase

Three e-libraries were selected to conduct this systematic review; Science-Direct (Elsevier); Springer-Link (Springer); and IEEE-Explore (IEEE). All are Scopus-indexed, constituting the biggest database of peer-reviewed research papers. The search was conducted directly in the selected e-libraries during 2017-2021, using the search keywords shown in Table 1.

Table 1. Keyword searching.

Keywords	Close Keywords	AND/OR Combination
NIDS	Network Intrusion Detection System	NIDS AND machine learning NIDS AND dimensionality reduction NIDS AND machine learning AND (feature selection OR feature extraction) NIDS AND machine learning AND dimensionality reduction
Machine Learning	Artificial Intelligence Techniques	
Feature Selection	Optimization Algorithms	
Feature Transformation	Feature Extraction	
Dimensionality Reduction	------	

4.2.2 Selecting Pre-processing Phase

The initial search process using the chosen keywords resulted in many initial hits, the titles of which were then cross-checked with the research questions and inclusion and exclusion criteria, to eliminate 550 papers not directly related to machine learning-based NIDSs. All authors independently scan the resulted 550 research papers (titles and abstracts). The resulting group was categorized into unrelated research papers (NR), partially related (PR) and related research papers (R). In this stage, the process of exclusion was performed on research papers the abstracts of which did not mention any techniques for supervised ML NIDS classification, feature selection, feature transformation and dimensionality reduction. A total of 170 research papers were marked NR and PR by the first review and then another review was performed on the unmarked set to judge 44 R papers. Further reviewing, to avoid any bias, was conducted. All reviewers met later to verify the exclusion of research papers deemed NR and PR. The final set of research papers was approved by all reviewers as related to this study, as shown in Figure 2. A total of 34 research papers remained and finally, the quality assessment criteria were followed again during the final full-text analysis for a total of 34 research papers.

4.2.3 Quality Assessment

The quality assessment eliminated bias in research papers selection and ensured that clear criteria were used to determine the quality of the selected research papers, as shown in Table 2. Scores for quality relied upon the following criteria: score 1 indicates that a research paper explicitly follows the assessment criteria, score 0 indicates that a research paper no doubt did not meet the criteria and research papers suspected to be related that necessitated more analysis and clarification or which did not fully meet the criteria were scored 0.5. Section 5 analyzes the papers that achieved over 50% in the quality assessment in detail.

Table 2. Quality assessment criteria.

	Assessment Question	**Assessment**
Q1	Does the paper topic cover NIDS domain?	1/ Zero/ 0.5
Q2	Does the paper use “machine learning techniques” or “machine learning and optimization techniques” or “machine learning and dimensionality reduction techniques”?	1/ Zero/ 0.5
Q3	Is the proposed methodology fully defined?	1/ Zero/ 0.5
Q4	Are the research results verified by clearly defined evaluation metrics?	1/ Zero/ 0.5

4.2.4 Information Extraction

The research questions require extracting information from the selected research papers, such as the use of: ML classification algorithms; dimensionality reduction techniques; and evaluation metrics and their results.

5. Supervised ML and Dimensionality Reduction Techniques

Answering RQ1 requires a complete analysis of the most popular supervised ML techniques (their implementation and algorithms) used to build NIDSs and detailed analysis of the dimensionality reduction techniques used.

5.1 Building Supervised ML NIDSs

Supervised ML provides an intelligence technique to extract patterns from previously labelled datasets [11], learning from previous datasets to predict future values [12]. Studies built NIDSs through several phases, including data pre-processing, training and testing and evaluation.

5.1.1 Data Pre-processing Phase

Dataset intensive care is required in supervised ML NIDSs to achieve the highest prediction accuracy rate and the most efficient performance in real-time intrusion detection; higher data quality indicates more NIDS efficiency [1]. Data pre-processing stages depend on dataset and ML algorithm requirements and researcher experience [1], [13]-[15]. In dataset cleaning, all duplicated or missing values are handled; duplicate values are deleted and rows with missing values may be deleted or filled with median, mean or most frequent corresponding values. Tables 3-5 show the research paper results for the ScienceDirect, IEEE and Springer-Link databases (respectively).

Table 3. ScienceDirect research paper results.

Research Paper	**Year**	**Q1**	**Q2**	**Q3**	**Q4**	**Total of 4**	**Percentage**
Nazir and Khan [32]	2021	1	1	1	0.5	3.5	87.5%
Mohammadi et al. [38]	2019	1	1	0.5	1	3.5	87.5%
Mazini *et al.* [33]	2019	1	1	0.5	1	3.5	87.5%
Alzahrani *et al.* [4]	2019	0.5	1	1	1	3.5	87.5%
Aljawarneh *et al.* [19]	2017	1	1	1	1	4	100%
Verma and Ranga [54]	2018	1	0.5	1	0.5	3	75%
Dwivedi *et al.* [39]	2020	0	1	1	1	3	75%
Shekhawat *et al.* [8]	2019	0.5	1	1	0.5	3	75%
Dahiya and Srivastava [10]	2018	1	1	1	1	4	100%
Kanimozhi and Jacob [40]	2020	1	1	1	0.5	3.5	87.5%
Hamamoto *et al.* [35]	2019	1	1	0.5	1	3.5	87.5%
Torres *et al.* [36]	2021	1	1	1	1	4	100%

Table 4. IEEE research paper results.

Research Paper	Year	Q1	Q2	Q3	Q4	Total of 4	Percentage
Vijayanand and Devaraj [18]	2020	1	1	1	1	4	100%
Stiawan *et al.* [16]	2020	1	1	1	1	4	100%
Jiang *et al.* [41]	2019	0.5	1	0.5	1	3	75%
Xue and Wu [17]	2020	1	1	1	0	3	75%
Ding *et al.* [42]	2020	1	1	1	0.5	3.5	87.5%
Nagaraja *et al.* [43]	2020	1	1	1	1	4	100%
Chang *et al.* [25]	2017	0.5	1	0.5	1	3	75%
Matel *et al.* [26]	2019	1	1	0.5	1	3.5	87.5%
Sun *et al.* [27]	2018	0.5	1	1	1	3.5	87.5%
Sakr *et al.* [34]	2019	1	1	1	1	4	100%

Table 5. Springer-Link research paper results.

Research Paper	Year	Q1	Q2	Q3	Q4	Total of 4	Percentage
Ghazy *et al.* [44]	2018	0.5	0.5	1	1	3	75%
Kunhare *et al.* [45]	2020	0.5	1	1	0.5	3	75%
Kasongo and Sun [46]	2020	0.5	1	1	0.5	3	75%
Bindra and Sood [47]	2019	0.5	1	1	1	3.5	87.5%
Rajadurai and Gandhi [48]	2020	0.5	1	1	0.5	3	75%
Alamiedy *et al.* [49]	2019	0.5	0.5	1	1	3	75%
Zhu and Zheng [50]	2019	0.5	0.5	1	1	3	75%
Sebbar *et al.* [51]	2020	0.5	1	1	1	3.5	87.5%
Thakur and Kumar [52]	2020	0.5	1	1	1	3.5	87.5%
Abhale and Manivannan [53]	2020	0.5	1	1	1	3.5	87.5%
Verma and Ranga [54]	2019	1	1	1	1	4	100%
Moon *et al.* [55]	2017	0.5	0.5	1	1	3	75%

All string values are transformed into numeric values, to be in a suitable format for the classification algorithm. For feature selection, unnecessary features are dropped either manually [14] or automatically (using dimensionality reduction techniques, as explained below). Some ML algorithms require normalization (data scaling) to ensure a uniform range between values (e.g. K-NN algorithm). Data splitting is applied by splitting datasets' columns and rows: columns are split into X, comprising all columns with independent variables; and Y is the column of the dependent variable that classifies rows (normal or abnormal traffic), called the "label column," which is the key data classification element in supervised learning.

5.1.2 Training Phase

To make the supervised ML algorithm goes through the learning experiment; it needs a partition of the dataset, called the training set [16]. The supervised ML classifier is fed with the independent (X) and dependent (Y) variables in the training set, to be able to predict Y values on its own in the future [12]. The size of the training set is important to help the ML algorithm learn efficiently with a highly accurate prediction rate in the least amount of time [17]. Most commonly, the training set consists of 70-80% of the original dataset, with the remainder for the testing set [16].

5.1.3 Testing and Evaluation Phase

The testing set is fed to the trained ML algorithm with only the X values, to test its ability to predict Y values. Predicted and actual Y values are then compared using evaluation metrics [16] (Section 6), to measure the trained ML algorithm's prediction ability and test its suitability with real NW traffic [18]. Supervised ML classification algorithms thus use independent (X) and dependent (Y) values and learn how they relate to each other in the training phase, then the trained algorithm is provided X values to evaluate performance in predicting Y values in the testing phase. Finally, the predicted results are evaluated using evaluation metrics [12].

5.2 Supervised ML Classifiers

5.2.1 Decision Tree (DT)

DT algorithm represents the feature values as nodes in a hierarchal tree, to divide the classification problems into sub-sets [19]. DT consists of nodes that represent features, branches represent roles and leaves represent a class value (e.g. malicious or normal traffic) [12]. DT algorithm forecasts class values based on learning decision rules extracted from features [20]. DT algorithm may be implemented by C4.5 (J48), an open-source Java implementation [21]; ID3, an extension of the former and REP-Tree [22], another open-source implementation for DT [23]. Aljawarneh *et al.* [19] proposed anomaly ML NIDS using REPTree classifier, pre-processed with feature selection using Vote scheme, training and testing phases. Their proposed NIDS obtained highly accurate results for detecting NW intrusions.

5.2.2 K-Nearest Neighbour (K-NN)

K-NN algorithm represents the given training data as neighbour points in a graph and assigns the new data point to the nearest specified K neighbour points. Figure 3 shows K-NN performance with K= 5. The distance between the new data point (X1, Y2) and any other neighbour point (X2, Y2) is calculated using Manhattan (Eq. 1) or Euclidean (Eq. 2) equations [1]. After calculating the distances, the new data point is classified according to the closest points [12].

$$|X2 - X1| + |Y2 - Y1| \tag{1}$$

$$\sqrt{((X2 - X1)^2 + (Y2 - Y1)^2)} \tag{2}$$

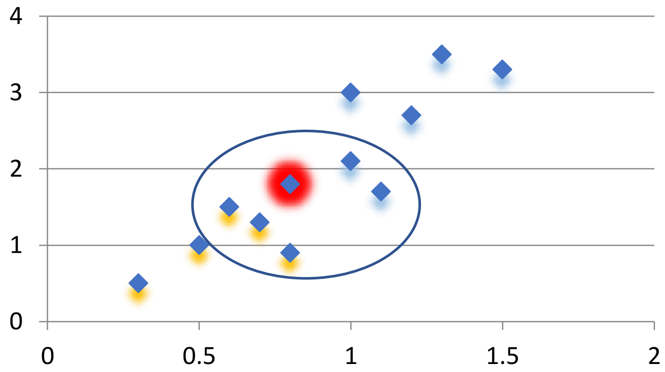


Figure 3. K-NN graph.

Verma and Ranga [7] used CIDDS-001 dataset to build their ML NIDS, labelled based on flow level and having 14 features; only 12 features were manually selected for the supervised training phase. Using the Weka tool, a K-NN classifier was implemented for multi-class classification. During the experiment, several K-NN iterations were conducted with different values for the number of neighbours (NN), identifying the best classification performance for NN = 2, with accuracy of 100% and no false positives.

5.2.3 Naïve Bayes (NB)

It is one of the most common machine learning classifiers in general. NB classification is based on Bayes' theorem [24]. NB measures the likelihood of a given prediction based on available features, as each feature independently contributes to predicting unknown data [2].

5.2.4 Support Vector Machine (SVM)

SVM algorithm combines statistical theory with supervised learning by finding the best way to split data into two classes by adding a boundary between them, regardless of whether the data can be divided linearly or not [8]. Essentially, this algorithm finds the best possible boundaries in the data collection to distinguish between classes [24].

5.2.5 ML Ensemble Methods

Ensemble supervised ML classifiers are integrated to solve a complex problem and increase accuracy by pooling individual classifiers' strengths [20]. For example, some algorithms may perform well in detecting a certain type of attack, but poorly in detecting others, thus combinations form a stronger classifier [25]. Several ML techniques (Random Forest (RF), Ada-Boost, XG-Boost, …etc.) use ensemble method to enhance performance. RF classifier integrates many DT classifiers, instead of depending on a single decision tree, taking predictions from each tree to forecast final performance based on the majority vote of predictions [17]. AdaBoost improves the performance of binary classifiers by employing an iterative approach, learning from the errors of weak classifiers and transforming them into strong ones [26]. XG-Boost consists of multiple DTs to solve a wide range of data-mining problems quickly and accurately [27].

5.3 Dimensionality Reduction Techniques

Supervised ML and Big Data mining techniques are very complex and require high computational costs due to the voluminous data processed [1]. Real-time detection and accurate detection rates in NIDSs are major concerns in relation to the “dimensional curse,” referring to ML model complexity due to a large number of both necessary and unnecessary features, with high dimensionality [28]. Dimensionality reduction techniques seek to reduce the number of features processed by selecting or extracting only relevant ones from the feature set, excluding irrelevant, noisy or redundant ones [29]. For dimensionality reduction, several algorithms reduce feature space either by removing features that do not provide important information or extracting relationships between available features to produce less space with new features [30]. This reduces complexity, increases understanding of data, facilitates easier analysis, improves visualization and reduces processing costs and storage space requirements [6], [29]. The ML model learning process is thus enhanced, resulting in higher performance and prediction accuracy rates, providing real-time prediction results [30]. Dimensionality reduction can be conducted by two approaches.

First, feature transformation/ extraction transforms the available features into more beneficial ones using optimization algorithms [28]. The most common methods used to conduct feature extraction are Principal Component Analysis (PCA), Multi-Dimensional Scaling (MDS), Isometric Mapping (ISOMAP), Locally Linear Embedding (LLE), Linear Discriminant Analysis (LDA), Canonical-Correlation-Analysis (CCA), Latent Semantic Indexing (LSI) and clustering methods [29].

Second, feature selection approach selects features according to their relevance and effectiveness related to the classification problem [29], without changing representation [31].

Researchers can choose one of four methods to implement their feature selection approach, which differ in how the ML algorithm functions [31] (Figure 4), as discussed below.

5.3.1 Filter Method

Weights are assigned to features to determine their relevance and essence (dependency, consistency, …etc.) using statistical standards, without involving the ML algorithm [29]. Depending on the assigned weights, features are either discarded or retained [31]. Filter method has been found to outperform other feature selection methods, with less computational costs, more scalability in high-dimensional datasets and more efficiency [29], [32]. Its drawbacks are that it does not integrate between the selected subset and the ML algorithm [29] and it is only suited to independent features [32].

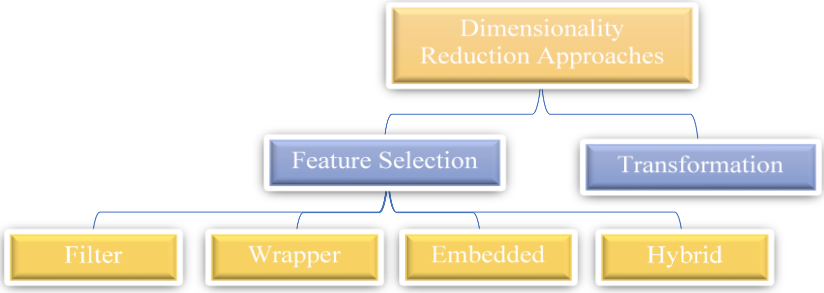


Figure 4. Dimensionality reduction techniques.

5.3.2 Wrapper Method

Wrapping creates an interaction between the ML algorithm later used for classification and each selected feature subset. The ML algorithm is used with each subset designated as a black box, to evaluate prediction accuracy and determine which subset has the fewest errors [30]. It is thus accurate and efficient [31], but is time-consuming, as selected subsets work only with particular ML algorithms, which may cause over-fitting, as well as being expensive [29].

5.3.3 Embedded Method

Embedding feature selection with the ML algorithm assigns weights independently and the highly weighted features are recursively used to construct subsets until finding the optimal one; its prediction accuracy outperforms others with the ML algorithm [30], [31]. The embedded method reduces the computational cost and the possibility of over-fitting [29].

5.3.4 Hybrid Method

The hybrid combination of filter and wrapper methods is the most commonly used solution, accruing the constituent advantages to achieve better performance [29]. This systematic review noticed that adopted dimensionality reduction techniques vary according to the research paper problem. In some problems, feature selection was forbidden, as removing features from the dataset would be misleading. Others preferred feature selection techniques, to keep meaningful original features and shorten the dimensionality reduction techniques within the selected features. Mazini *et al.* [33] proposed hybrid anomaly NIDS to detect attacks that threaten network activities. They mentioned that data-mining techniques were implemented to get rid of imbalanced database disadvantages and the complicity of feature values. Furthermore, to reach the best performance of the AdaBoost classification algorithm, they used the Artificial Bee Colony algorithm (the wrapper method) for feature selection. Selecting the most significant features to learn the classifier increases accuracy detection rate and reduces false alarms.

5.4 RQ1 Analysis and Results

Answering RQ1 entails understanding ML techniques, the most commonly used supervised ML classifiers and dimensionality reduction techniques, as summarized in Tables 6-8. Some of the selected research papers used ML classification algorithms to build their NIDS, while others additionally used dimensionality reduction technique, to improve prediction results and increase NIDS sensitivity and accuracy.

Table 6. ML methods and dimensionality reduction techniques employed – ScienceDirect studies.

Research Paper	**Training ML Algorithms**	**Feature Selection**			**FT**	**Dataset**
		Fil	Wra	Emb		
Nazir and Khan [32]	RF		✓			UNSW-NB15
Mohammadi *et al.* [38]	DT, Least Square SVM	✓	✓			KDD CUP 99
Mazini *et al.* [33]	Adaboost		✓			NSL-KDD, ISCXIDS2012
Alzahrani *et al.* [4]	SVM		✓			NSL-KDD
Aljawarneh *et al.* [19]	RF, J48, AdaBoost, NB	✓				NSL-KDD
Verma and Ranga [54]	K-NN					CIDDS-001
Dwivedi *et al.* [39]	SVM	✓	✓			ISCX 2012, NSL-KDD, CIC-IDS2017
Shekhawat *et al.* [8]	RF, SVM, XG-boost					CTU-13, Malware Capture Facility Project dataset
Dahiya and Srivastava [10]	RF, REP TREE, NB				✓	UNSW-NB15
Kanimozhi and Jacob [40]	RF, SVM, NB, K-NN, AdaBoost with DT		✓			CSE-CIC-IDS2018
Hamamoto *et al.* [35]	RF					Private dataset
Torres *et al.* [36]	RF	✓	✓			Private dataset

Fil: filter; Wra: wrapper; Emb: embedded; FT: feature transformation.

Table 7. ML methods and dimensionality reduction techniques employed – IEEE studies.

Research Paper	Training ML Algorithms	Feature Selection			FT	Dataset
		Fil	Wra	Emb		
Vijayanand and Devaraj [18]	SVM, RF				✓	CICIDS2017, ADFA-LD
Stiawan *et al.* [16]	J48			✓		ITD-UTM
Jiang *et al.* [41]	XG-Boost, RF				✓	KDD99, NSL-KDD
Xue and Wu [17]	SVM, XG-Boost					Private Dataset
Ding *et al.* [42]	SVM, NB				✓	UNSW_NB15
Nagaraja *et al.* [43]	J48	✓				NSL-KDD
Chang *et al.* [25]	SVM, RF				✓	KDD99
Matel *et al.* [26]	SVM		✓			DARPA KDD CUP 99
Sun *et al.* [27]	SVM			✓		KDD CUP 99
Sakr *et al.* [34]	SVM				✓	NSL-KDD

Filt: filter; Wra: wrapper; Emb: embedded; FT: feature transformation.

Table 8. ML methods and dimensionality reduction techniques employed – Springer-Link studies.

Research Paper	Training ML Algorithms	Feature Selection			FT	Dataset
		Fil	Wra	Emb		
Ghazy *et al.* [44]	RF		✓		✓	NSL-KDD
Kunhare *et al.* [45]	RF			✓		NSL-KDD
Kasongo and Sun [46]	XG-Boost- DT	✓				UNSW-NB15
Bindra and Sood [47]	RF				✓	CIC IDS 2017
Rajadurai and Gandhi [48]	Ensemble Gradient descent, RF			✓		NSL-KDD
Alamiedy *et al.* [49]	RF		✓			NSL–KDD
Zhu and Zheng [50]	SVM					Private dataset
Sebbar *et al.* [51]	RF	✓				Private dataset
Thakur and Kumar [52]	RF	✓	✓			Private Dataset
Abhale and Manivannan [53]	SVM					NSL-KDD
Verma and Ranga [54]	DT				✓	RPL-NIDDS17
Moon *et al.* [55]	DT					Private dataset

Filt: filter; Wra: wrapper; Emb: embedded; FT: feature transformation.

6. EVALUATION METRICS

Answering RQ2 and RQ3 requires a complete analysis of evaluation metrics used to evaluate the proposed NIDS in each research paper.

6.1 Evaluation Metrics

During the building of any ML model, particularly in the testing phase, many metrics are used to evaluate performance [7], [27], [32]. Most of these measures are derived from the confusion matrix, which consists of two columns displaying predicted values and two rows displaying the actual values. In NIDS, predicted or actual values are positive if NW traffic is positive or negative if normal, as shown in Figure 5 [1].

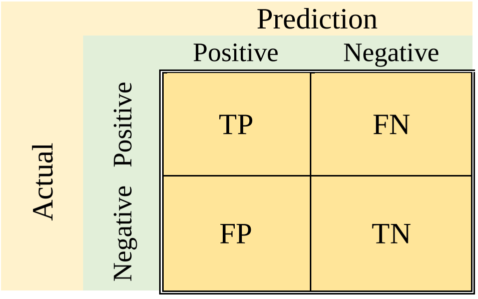


Figure 5. Confusion matrix.

Each intersection between the columns and rows contains the following values [7]: True Positives (TP): the number of values predicted as attacks that are attacks; False Negatives (FN): the number values predicted as normal traffic that are attacks; False Positives (FP): the number of values predicted as attacks that are normal traffic; and True Negatives (FN): the number of values predicted as normal that are normal traffic. The evaluation metrics from the confusion matrix used to evaluate the proposed NIDS varied between those discussed below.

Accuracy Rate (AR) and Error Rate (ER) recognize intrusions, indicated by the ratio of correctly predicted values (TP and TN) to all other values (Eq. 3.a) [35]. ER is calculated depending on the AR, as shown in Eq. 3.b.

$$Accuracy\ rate\ (AR) = \frac{TP+TN}{TP+FN+FP+TN} \tag{3.a}$$

$$Error\ rate\ (ER) = 100 - AR \tag{3.b}$$

Recall Value (Re-V) (detection rate) measures NIDS sensitivity [12], [36]. It is the ratio of correctly predicted values as attacks (TP) to all other values that are in fact attacks (Eq. 4) [1].

$$Recall\ value = \frac{TP}{TP+FN} \tag{4}$$

Precision value (PV) indicates the reliability of the NIDS [12]. It is the ratio of correctly predicted values as attacks (TP) to all other predicted values as attacks (Eq. 5) [1].

$$Precision = \frac{TP}{TP+FP} \tag{5}$$

False alarm rate is the ratio of incorrectly predicted values as attacks (FP) to all other normal values (Eq. 6) [1].

$$False\ alarm\ rate\ = \frac{FP}{FP+TN} \tag{6}$$

True Negative Rate (TNR) measures NIDS specificity [13]; it is the ratio of correctly predicted values as normal traffic (TN) to all other normal values (Eq. 7) [1].

$$True\ negative\ rate\ = \frac{TN}{FP+TN} \tag{7}$$

F Measure (F1) represents NIDS accuracy in terms of precision and recall values (Eq. 8) [13].

$$F\ Measure\ = 2\left(\frac{Precession\ X\ Recall}{Precession + Recall}\right) \tag{8}$$

Receiver Operating Character – Area Under the Curve (Roc-Auc) rate is the area under the curve that virtualizes the relation between the True Positive Rate (TPR) and False Positive Rate (FPR) for every confusion matrix, resulting from every threshold in binary classification [8], [37]. The higher the TPR and the lower the FPR, the higher the Roc-Auc score [13]. For further evaluation of the NIDS performance, researchers calculate the time consumed in the training and testing phases (Tr-T and Ts-T, respectively), so the NIDS is lightweight and easy to install and provides real-time detection of NW intrusions [12]. Tables 9-11 show that most researchers relied on AR, DR and FPR to evaluate their proposed NIDS, so these metrics are considered to answer RQ3 in the next section.

Table 9. Results of evaluation metrics for each research paper – ScienceDirect.

Research Paper	**Evaluation Metrics**							
	AR %	Re-V/DR %	PV	FPR%	F1	Auc-Roc	Tr-T (sec)	Ts-T (sec)
Nazir and Khan [32]	83.12			3.7				
Mohammadi *et al.* [38]	95.03	95.23		1.65				
Mazini *et al.* [33]	98.9	99.61		1				
Alzahrani *et al.* [4]		99.21					0.6385	
Aljawarneh *et al.* [19]	99.81			0.3				
Verma and Ranga [54]	100							
Dwivedi *et al.* [39]	99.63	99.71		8.5				
Shekhawat *et al.* [8]	100					99.88		
Dahiya and Srivastava [10]	93.56	0.843	84.2	2.1		96.1	5.74	
Kanimozhi and Jacob [40]	99.96	99.88	99.96		99.92	99.88		
Hamamoto *et al.* [35]	96.53			0.56				
Torres *et al.* [36]	93			7.5	93	97		

Table 10. Results of evaluation metrics for each research paper – IEEE.

Research Paper	**Evaluation Metrics**							
	AR %	Re-V/DR %	PV	FPR%	F1	Auc-Roc	Tr-T (sec)	Ts-T (sec)
Vijayanand *et al.* [18]	95.91			4			4959	4960
Stiawan *et al.* [16]	99.87						0.996	0.830
Jiang *et al.* [41]	94	0.75	81.0		0.71			
Xue and Wu [17]	99.68							
Ding *et al.* [42]	99.41	99.64	99.04	0.0077			144.3	2.39
Nagaraja *et al.* [43]	99.44	87.6	92.5			.981		
Chang *et al.* [25]	93			3				
Matel *et al.* [26]	96.122			3.878				
Sun *et al.* [27]	91.686							
Sakr *et al.* [34]	98.04	97.55		1.4			5.16	10.18

Table 11. Results of evaluation metrics for each research paper – Springer-Link.

Research Paper	**Evaluation Metrics**							
	AR %	Re-V/DR %	PV	FPR%	F1	Auc-Roc	Tr-T (sec)	Ts-T (sec)
Ghazy *et al.* [44]			0.99	0.001				
Kunhare *et al.* [45]	99.32	99.26	99.37	0.62	99.31			
Kasongo and Sun [46]	90.85		80.33		87.42			
Bindra and Sood [47]	96					0.99		
Rajadurai and Gandhi [48]	91.06	99.77						
Alamiedy *et al.* [49]	93.64							
Zhu and Zheng [50]	99.31							
Sebbar *et al.* [51]	97.4	98.9	94.7		96.7	99		
Thakur and Kumar [52]	99.1							
Abhale and Manivannan [53]	84.0	0.86	0.87	0.8	0.87	0.85		
Verma and Ranga [54]	94.07			3.80				
Moon *et al.* [55]	89.1	84.7						

6.2 RQ2 and RQ3 Results

For RQ2, after analyzing the used evaluation metrics to determine the highest evaluation results achieved by the selected research papers, Tables 12-14 show that most researchers relied on AR, DR and FPR to evaluate their proposed NIDS, so these metrics are considered to answer RQ3. Determining the best ML and dimensionality reduction techniques used to build NIDS requires summarizing all techniques used in the selected research papers, showing their AR, DR and FPR (Tables 12-14).

Table 12. Summary of all techniques used in the selected research papers – ScienceDirect.

Research Paper	**ML Algorithms**	**Feature Selection**			**FT**	**Evolution Metric**		
		Fil	Wra	Emb		AR	DR	FPR
Nazir and Khan [32]	RF	✓	✓			83.12		3.7
Mohammadi *et al.*	DT	✓	✓			95.03	95.23	1.65
Mazini *et al.* [33]	Ada-Boost		✓			98.9	99.61	1
Alzahrani *et al.* [4]	SVM		✓				99.21	
Aljawarneh *et al.* [19]	RF, J48, Ada-Boost, NB	✓				99.81		0.3
Verma and Ranga [54]	K-NN					100		
Dwivedi *et al.* [39]	SVM	✓	✓			99.63	99.71	8.5
Shekhawat *et al.* [8]	RF, SVM, XG-boost					100		
Dahiya and Srivastava	RF, REP TREE, NB				✓	93.56	84.3	2.1
Kanimozhi and Jacob	RF, SVM, NB,		✓			99.96	99.88	
Hamamoto *et al.* [35]	RF					93		7.5
Torres *et al.* [36]	RF	✓	✓			98.92		

Fil: filter; Wra: wrapper; Emb: embedded; FT: feature transformation.

The ML algorithm (marked in red) indicates its adoption in the research paper, achieving the best evaluation results. The tables also show that most researchers relied on AR, DR and FPR to evaluate their proposed NIDS, so these metrics are considered to answer RQ3.

Table 13. Summary of all techniques used in the selected research papers – IEEE.

Research Paper	**ML Algorithms**	**Feature Selection**			**FT**	**Evolution Metric**		
		Fil	Wra	Emb		AR	DR	FPR
Vijayanand *et al.* [18]	SVM, RF				✓	95.91		4
Stiawan *et al.* [16]	J48			✓		99.87		
Jiang *et al.* [41]	XGBoost, RF				✓	94	0.75	
Xue and Wu [17]	SVM, XGBoost				✓	99.68		
Ding *et al.* [42]	SVM, NB	✓				99.41	99.64	0.77
Nagaraja *et al.* [43]	J48				✓	99.44	87.6	
Chang *et al.* [25]	SVM, RF		✓					3
Matel *et al.* [26]	SVM			✓		93		3.878
Sun *et al.* [27]	SVM				✓	96.122		
Sakr *et al.* [34]	SVM	✓	✓		✓	91.686	97.55	1.4

Filt: filter; Wra: wrapper; Emb: embedded; FT: feature transformation.

Table 14. Summary of all techniques used in the selected research papers – Springer-Link.

Research Paper	**ML Algorithms**	**Feature Selection**			**FT**	**Evolution Metric**		
		Fil	Wra	Emb		AR	DR	FPR
Ghazy *et al.* [44]	RF		✓		✓			0.001
Kunhare *et al.* [45]	RF			✓		99.32	99.26	0.62
Kasongo and Sun [46]	XGBoost- DT	✓				90.85		
Bindra and Sood [47]	RF				✓	96		
Rajadurai and Gandhi [48]	Ensemble Gradient descent, RF			✓		91.06	99.77	
Alamiedy *et al.* [49]	RF		✓			93.64		
Zhu and Zheng [50]	SVM					99.31		
Sebbar *et al.* [51]	RF	✓				97.4	98.9	
Thakur and Kumar [52]	RF	✓	✓			99.1		
Abhale and Manivannan [53]	SVM					84.0	0.86	0.8
Verma and Ranga [54]	DT				✓	94.07		3.80
Moon *et al.* [55]	DT					89.1	84.7	

Filt: filter; Wra: wrapper; Emb: embedded; FT: feature transformation.

7. Discussion and Future Challenges

7.1 Main Findings

Figure 6 shows how many supervised ML classifiers are used in the selected research papers. It can be observed that RF classifier is generally preferred, due to its accurate classification performance (i.e., ability to detect zero-day attacks) and low computational costs in real time (Table 6). From the selected research papers, feature selection is the most used dimensionality reduction technique for the proposed NIDS (Figure 7). These techniques reduce feature dimensionality to reduce the complexity of the training and testing phases, ultimately ensuring real-time detection, but at the cost of more computational resources. Figure 8 shows that the most used evaluation metrics are AR, DR and FPR. Efficient NIDS requires high AR and DR, with low FPR. Thus, to evaluate the efficiency of the NIDS, these values must be calculated.

7.2. Research Challenges

Most of the proposed NIDSs were constructed in laboratory conditions (not in a real environment), using predefined datasets and there is no proof of their efficiency in real-world implementations. Testing NIDS effectiveness in real NW traffic remains a research challenge. The proposed NIDS is complex and its computational and time costs are considerable, which may affect real-time detection. Although dimensionality reduction techniques are being used for this purpose, more improvement is still needed in the field.

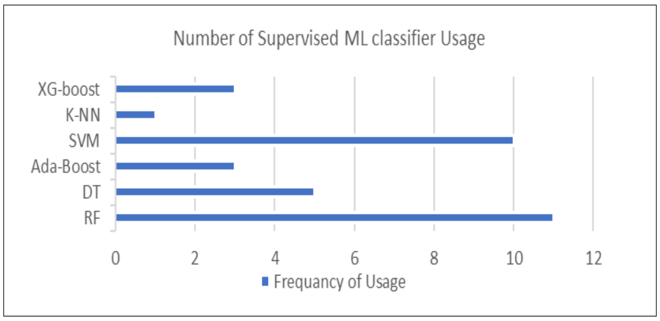
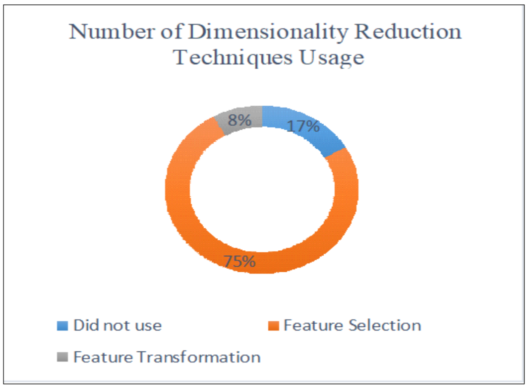


Figure 6. Use of supervised ML classifier.



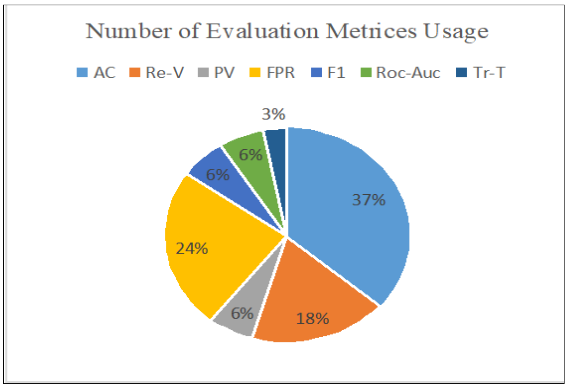


Figure 7. Use of dimensionality reduction techniques.

Figure 8. Use of evaluation metrics.

8. Conclusions

This systematic review extensively analyzed NIDSs based on supervised ML classifiers and dimensionality reduction techniques to provide updated knowledge for new interested researchers in this field. A systematic approach was adopted to select relevant research papers to answer the RQs. According to the results, RF is the most supervised ML classifier, due to its accurate classification performance and low computational costs. Feature selection techniques are the most used for dimensionality reduction in recently proposed NIDSs. These techniques reduce feature dimensionality to reduce the complexity of the training and testing phases and eventually ensure accurate real-time detection, but they need more computational resources. The most commonly used metrics are AR, DR and FPR. An efficient NIDS requires high AR and DR, with low FPR; these values must be determined for NIDS efficiency evaluation. This systematic review concludes that despite all efforts in the ML NIDS field, there are still some challenges facing interested researchers, including proving the effectiveness of the proposed ML NIDS implementation in a real NW traffic environment and reducing its complexity to ensure real-time detection. This systematic review is limited by being restricted to only 34 research papers within the domain of supervised anomaly ML-based NIDSs. Future work needs to address more research papers in a broader domain, including ML and deep learning techniques.

References

[1] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah and F. Ahmad, "Network Intrusion Detection System: A Systematic Study of Machine Learning and Deep Learning Approaches," Transactions on Emerging Telecom. Technologies, vol. 32, no. 1, pp. 1–29, DOI: 10.1002/ett.4150, Oct. 2021.

[2] P. Spadaccino and F. Cuomo, "Intrusion Detection Systems for IoT: Opportunities and Challenges Offered by Edge Computing," *arXiv*, Art no. 2012.01174v1, Dec. 2020.

[3] D. Anderson, T. Frivold and A. Valdes, "Next-generation Intrusion Detection Expert System (NIDES): A Summary," Computer Science Laboratory Rep. SRI-CSL-95-07, [Online], Available: http://merlot.usc.edu/cs530-s04/papers/Anderson95a.pdf, May 1995.

[4] A. S. Alzahrani, R. A. Shah, Y. Qian and M. Ali, "A Novel Method for Feature Learning and Network Intrusion Classification," Alexandria Engineering Journal, vol. 59, no. 3, pp. 1159–1169, Jun. 2020.

[5] W. A. Gould, "Spoilage of Canned Tomatoes and Tomato Products," in: Tomato Production, Processing and Technology, Ch. 25, pp. 419–431, 3rd Ed., Sawston, U.K.: Woodhead Publishing, DOI: 10.1533/9781845696146.3.419, 1992.

[6] S. M. Othman, F. Mutaher Ba-Alwi, N. T. Alsohybe and A. T. Zahary, "Survey on Intrusion Detection System Types," Int. J. Cyber-Security Digit. Forensics, vol. 7, no. 4, pp. 444–462, Dec. 2018.

[7] A. Verma and V. Ranga, "Statistical Analysis of CIDDS-001 Dataset for Network Intrusion Detection Systems Using Distance-based Machine Learning," Procedia Computer Science, vol. 125, pp. 709–716, DOI: 10.1016/j.procs.2017.12.091, Dec. 2018.

[8] A. S. Shekhawat, F. Di Troia and M. Stamp, "Feature Analysis of Encrypted Malicious Traffic," Expert Systems with Applications, vol. 125, pp. 130–141, DOI: 10.1016/j.eswa.2019.01.064, Feb. 2019.

[9] T. Aldwairi, D. Perera and M. A. Novotny, "An Evaluation of the Performance of Restricted Boltzmann Machines As a Model for Anomaly Network Intrusion Detection," Computer Networks, vol. 144, pp. 111–119, DOI: 10.1016/j.comnet.2018.07.025, Oct. 2018.

[10] P. Dahiya and D. K. Srivastava, "Network Intrusion Detection in Big Dataset Using Spark," Procedia Computer Science, vol. 132, pp. 253–262, DOI: 10.1016/j.procs.2018.05.169, 2018.

[11] A. Singh, N. Thakur and A. Sharma, "A Review of Supervised Machine Learning Algorithms," Proc. of the 3rd IEEE International Conference on Computing for Sustainable Global Development (INDIACom), pp. 1310–1315, New Delhi, India, Mar. 2016.

[12] S. Khatri, A. Arora and A. P. Agrawal, "Supervised Machine Learning Algorithms for Card Fraud Detection: A Comparison," Proc. of the 10th IEEE Int. Conf. on Cloud Computing, Data Science &Amp; Engineering (Confluence), pp. 680–683, Noida, India, Mar. 2020.

[13] M. Qasaimeh, R. Turab and R. S. Al-Qassas, "Authentication Techniques in Smart Grid: A Systematic Review," Telkomnika, vol. 17, no. 3, pp. 1584–1594, DOI: 10.12928/TELKOMNIKA.V17I3.11437, Jun. 2019.

[14] Z. Ashi, L. Aburashed, M. Al-Fawa'reh and M. Qasaimeh, "Fast and Reliable DDoS Detection Using Dimensionality Reduction and Machine Learning," Proc. of the 15th IEEE Int. Conf. for Internet Technology and Secured Transactions (ICITST), DOI: 10.23919/ICITST51030.2020.9351347, London, UK, Dec. 2020.

[15] S. B. Kotsiantis, D. Kanellopoulos and P. E. Pintelas, "Data Preprocessing for Supervised Learning," International Journal of Computer and Information Engineering's, vol. 1, no. 12, pp. 4104–4110, 2007.

[16] D. Stiawan et al., "An Approach for Optimizing Ensemble Intrusion Detection Systems," IEEE Access, vol. 9, pp. 6930–6947, DOI: 10.1109/ACCESS.2020.3046246, Dec. 2021.

[17] W. Xue and T. Wu, "Active Learning-based XGBoost for Cyber Physical System against Generic AC False Data Injection Attacks," IEEE Access, vol. 8, pp. 144575–144584, Aug. 2020.

[18] R. Vijayanand and D. Devaraj, "A Novel Feature Selection Method Using Whale Optimization Algorithm and Genetic Operators for Intrusion Detection System in Wireless Mesh Network," IEEE Access, vol. 8, pp. 56847–56854, DOI: 10.1109/ACCESS.2020.2978035, Mar. 2020.

[19] S. Aljawarneh, M. Aldwairi and M. B. Yassein, "Anomaly-based Intrusion Detection System through Feature Selection Analysis and Building Hybrid Efficient Model," J. of Computational Science, vol. 25, no. October, pp. 152–160, DOI: 10.1016/j.jocs.2017.03.006, Mar. 2018.

[20] C. T. Tran, M. Zhang, P. Andreae and B. Xue, "Bagging and Feature Selection for Classification with Incomplete Data," Proc. of the European Conference on the Applications of Evolutionary Computation (EvoApplications 2017), Part of the Lecture Notes in Computer Science Book Series, vol. 10199, Cham: Springer, DOI: 10.1007/978-3-319-55849-3_31, 2017.

[21] R. Ihya, A. Namir, S. El Filali, M. Ait Daoud and F. Z. Guerss, "J48 Algorithms of Machine Learning for Predicting a User's Acceptance of an E-orientation Systems," Proceedings of the 4th ACM International Conference on Smart City Applications (SCA '19), Article no. 20, pp. 1-8, DOI: 10.1145/3368756.3368995, Oct. 2019.

[22] F. Alam and S. Pachauri, "Comparative Study of J48, Naive Bayes and One-R Classification Technique for Credit Card Fraud Detection Using WEKA," Journal of Advanced Computer Science & Technology, vol. 10, no. 6, pp. 1731–1743, 2017.

[23] R. Harode, "XGBoost: A Deep Dive into Boosting," SFU/// Professional Master's Program in Computer Science, [Online], Available: https://medium.com/sfu-cspmp/xgboost-a-deep-dive-into-boosting-

f06c9c41349 (accessed on Oct. 15, 2021).

[24] M. Guia, R. R. Silva and J. Bernardino, "Comparison of Naive Bayes, Support Vector Machine, Decision Trees and Random Forest on Sentiment Analysis," Proc. 11th Int. Jt. Conf. Knowl. Discov. Knowl. Eng. Knowl. Manag. (KDIR 2019), vol. 1, pp. 525–531, DOI: 10.5220/0008364105250531, Nov. 2019.

[25] Y. Chang, W. Li and Z. Yang, "Network Intrusion Detection Based on Random Forest and Support Vector Machine," Proc. of 2017 IEEE Int. Conf. on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), vol. 1, pp. 635–638, DOI: 10.1109/CSE-EUC.2017.118, Guangzhou, China, Jul. 2017.

[26] E. C. Matel, A. M. Sison and R. P. Medina, "Optimization of Network Intrusion Detection System Using Genetic Algorithm with Improved Feature Selection Technique," Proc. of the IEEE 11th Int. Conf. Humanoid, Nanotechnology, Inf. Technol. Commun. Control. Environ. Manag. (HNICEM), Article no. 19556390, Laoag, Philippines, DOI: 10.1109/HNICEM48295.2019.9073439, 2019.

[27] S. Sun, Z. Ye, L. Yan, J. Su and R. Wang, "Wrapper Feature Selection Based on Lightning Attachment Procedure Optimization and Support Vector Machine for Intrusion Detection," Proc. of the 4th IEEE Int. Symposium on Wireless Systems within the Int. Conf. on Intelligent Data Acquisition and Advanced Computing Systems, pp. 41–46, Lviv, Ukraine, Sep. 2018.

[28] M. Pechenizkiy, A. Tsymbal and S. Puuronen, "PCA-based Feature Transformation for Classification: Issues in Medical Diagnostics," Proc. of the 17th IEEE Symposium on Computer-based Medical Systems, vol. 17, pp. 535–540, DOI: 10.1109/cbms.2004.1311770, Bethesda, MD, USA, Jul. 2004.

[29] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari and J. Saeed, "A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction," Journal of Applied Science and Technology Trends, vol. 1, no. 2, pp. 56–70, DOI: 10.38094/jastt1224, May 2020.

[30] D. Mladenić, "Feature Selection for Dimensionality Reduction," Proc. of Subspace, Latent Structure and Feature Selection, vol. 3940, C. Saunders, M. Grobelnik, S. Gunn and J. Shawe-Taylor, Eds., Berlin: Springer, pp. 84–102, DOI: 10.1007/11752790_5, 2006.

[31] M. Masaeli, G. Fung and J. G. Dy, "From Transformation-based Dimensionality Reduction to Feature Selection," Proceedings of the 27th International Conference on Machine Learning, pp. 751–758, DOI: 10.5555/3104322.3104418, Haifa, 2010.

[32] A. Nazir and R. A. Khan, "A Novel Combinatorial Optimization based Feature Selection Method for Network Intrusion Detection," Computers & Security, vol. 102, Article no. 102164, DOI: 10.1016/j.cose.2020.102164, Mar. 2021.

[33] M. Mazini, B. Shirazi and I. Mahdavi, "Anomaly Network-based Intrusion Detection System Using a Reliable Hybrid Artificial Bee Colony and AdaBoost Algorithms," Journal of King Saud University - Computer and Information Sciences, vol. 31, no. 4, pp. 541–553, Oct. 2019.

[34] M. M. Sakr, M. A. Tawfeeq and A. B. El-Sisi, "Filter *versus* Wrapper Feature Selection for Network Intrusion Detection System," Proc. of the IEEE 9th Int. Conf. Intell. Comput. Inf. Syst. (ICICIS), pp. 209–214, DOI: 10.1109/ICICIS46948.2019.9014797, Cairo, Egypt, Dec. 2019.

[35] A. H. Hamamoto, L. F. Carvalho, L. D. H. Sampaio, T. Abrão and M. L. Proença, "Network Anomaly Detection System Using Genetic Algorithm and Fuzzy Logic," Expert Systems with Applications, vol. 92, no. C, pp. 390–402, DOI: 10.1016/j.eswa.2017.09.013, Feb. 2018.

[36] J. L. G. Torres, C. A. Catania and E. Veas, "Active Learning Approach to Label Network Traffic Datasets," Journal of Information Security and Applications, vol. 49, Article no. 102388, 2019.

[37] B. Anderson and D. McGrew, "Machine Learning for Encrypted Malware Traffic Classification: Accounting for Noisy Labels and Non-stationarity," Proc. of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17), vol. F1296, pp. 1723–1732, DOI: 10.1145/3097983.3098163, Aug. 2017.

[38] S. Mohammadi, H. Mirvaziri, M. Ghazizadeh-Ahsaee and H. Karimipour, "Cyber Intrusion Detection by Combined Feature Selection Algorithm," Journal of Information Security and Applications, vol. 44, pp. 80–88, DOI: 10.1016/j.jisa.2018.11.007, Feb. 2019.

[39] S. Dwivedi, M. Vardhan and S. Tripathi, "An Effect of Chaos Grasshopper Optimization Algorithm for Protection of Network Infrastructure," Computers Networks, vol. 176, Article no. 107251, DOI: 10.1016/j.comnet.2020.107251, May 2020.

[40] V. Kanimozhi and T. P. Jacob, "Artificial Intelligence Outflanks All Other Machine Learning Classifiers in Network Intrusion Detection System on the Realistic Cyber Dataset CSE-CIC-IDS2018 Using Cloud Computing," ICT Express, vol. 7, no. 3, pp. 366–370, DOI: 10.1016/j.icte.2020.12.004, Sep. 2021.

[41] H. Jiang, Z. He, G. Ye and H. Zhang, "Network Intrusion Detection based on PSO-Xgboost Model," IEEE Access, vol. 8, pp. 58392–58401, DOI: 10.1109/ACCESS.2020.2982418, Mar. 2020.

[42] P. Ding, J. Li, M. Wen, L. Wang and H. Li, "Efficient BiSRU Combined with Feature Dimensionality Reduction for Abnormal Traffic Detection," IEEE Access, vol. 8, pp. 164414–164427, DOI: 10.1109/ACCESS.2020.3022355, Sep. 2020.

[43] A. Nagaraja, U. Boregowda, K. Khatatneh, R. Vangipuram, R. Nuvvusetty and V. Sravan Kiran, "Similarity-based Feature Transformation for Network Anomaly Detection," IEEE Access, vol. 8, pp. 39184–39196, DOI: 10.1109/ACCESS.2020.2975716, Feb. 2020.

[44] R. A. Ghazy, E. S. M. EL-Rabaie, M. I. Dessouky, N. A. El-Fishawy and F. E. Abd El-Samie, "Efficient Techniques for Attack Detection Using Different Features Selection Algorithms and Classifiers," Wireles Personal Communication, vol. 100, no. 4, pp. 1689–1706, DOI: 10.1007/s11277-018-5662-0, May 2018.

[45] N. Kunhare, R. Tiwari and J. Dhar, "Particle Swarm Optimization and Feature Selection for an Intrusion Detection System," Sadhana, vol. 45, Article no. 109, DOI: 10.1007/s12046-020-1308-5, May 2020.

[46] S. M. Kasongo and Y. Sun, "Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset," Journal of Big Data, vol. 7, Article no. 105, DOI: 10.1186/s40537-020-00379-6, Nov. 2020.

[47] N. Bindra and M. Sood, "Detecting DDoS Attacks Using Machine Learning Techniques and Contemporary Intrusion Detection Dataset," Automatic Control and Computer Sciences, vol. 53, no. 5, pp. 419–428, DOI: 10.3103/S0146411619050043, Nov. 2019.

[48] H. Rajadurai and U. D. Gandhi, "A Stacked Ensemble Learning Model for Intrusion Detection in a Wireless Network," Neural Comp. and App., vol. 5, DOI: 10.1007/s00521-020-04986-5, May 2020.

[49] T. A. Alamiedy, M. Anbar, Z. N. M. Alqattan and Q. M. Alzubi, "Anomaly-based Intrusion Detection System Using Multi-objective Grey Wolf Optimization Algorithm," Journal of Ambient Intelligence and Humanized Computing, vol. 11, pp. 3735–3756, DOI: 10.1007/s12652-019-01569-8, Nov. 2019.

[50] Y. Zhu and Y. Zheng, "Traffic Identification and Traffic Analysis Based on Support Vector Machine," Neural Comput. Appl., vol. 32, pp. 1903–1911, DOI: 10.1007/s00521-019-04493-2, Sep. 2020.

[51] A. Sebbar, K. Zkik, Y. Baddi, M. Boulmalf and M. D. E. C. El Kettani, "MitM Detection and Defense Mechanism CBNA-RF Based on Machine Learning for Large-scale SDN Context," J. of Ambient Intelligence and Humanized Comp., vol. 11, pp. 5875–5894, DOI: 10.1007/s12652-020-02099-4, 2020.

[52] K. Thakur and G. Kumar, "Nature Inspired Techniques and Applications in Intrusion Detection Systems: Recent Progress and Updated Perspective," Archives of Computational Methods in Engineering, Article no. 0123456789, DOI: 10.1007/s11831-020-09481-7, Aug. 2020.

[53] A. B. Abhale and S. S. Manivannan, "Supervised Machine Learning Classification Algorithmic Approach for Finding Anomaly Type of Intrusion Detection in Wireless Sensor Network," Optical Memory and Neural Networks, vol. 29, pp. 244-256, DOI: 10.3103/S1060992X20030029, 2020.

[54] A. Verma and V. Ranga, "Evaluation of Network Intrusion Detection Systems for RPL Based 6LoWPAN Networks in IoT," Wireless Personal Communications, vol. 108, pp. 1571–1594, DOI: 10.1007/s11277-019-06485-w, Oct. 2019.

[55] D. Moon, H. Im, I. Kim and J. H. Park, "DTB-IDS: An Intrusion Detection System Based on Decision Tree Using behavior Analysis for Preventing APT Attacks," Journal of Supercomputing, vol. 73, pp. 2881–2895, DOI: 10.1007/s11227-015-1604-8, Jul. 2017.

[56] N. Martins, J. M. Cruz, T. Cruz and P. H. Abreu, "Adversarial Machine Learning Applied to Intrusion and Malware Scenarios: A Systematic Review," IEEE Access, vol. 8, pp. 35403–35419, Feb. 2020.

[57] A. A. Ramaki, A. Rasoolzadegan and A. J. Jafari, "A Systematic Review on Intrusion Detection Based on the Hidden Markov Model," Statistical Analysis and Data Mining, vol. 11, pp. 111–134, Apr. 2018.

[58] C. Gonzalez, "Increasing Security in Military Self-protected Software," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 7, no. 3, pp. 253–267, Sep. 2021.

[59] A. B. Nassif, M. A. Talib, Q. Nasir, H. Albadani and F. M. Dakalbab, "Machine Learning for Cloud Security: A Systematic Review," IEEE Access, vol. 9, pp. 20717–20735, Jan. 2021.

ملخص البحث:

إنّ حماية السّرية والسلامة والتّوافُر للحيّز السّيبراني وأصول الشّبكات أصبحت مسألة تحظى بلاهتمام بشكلٍ متزايد. وقد خلقت الزّيادة السّريعة في استخدام الإنترنت ووجود الأنظمة الحديثة للحوسبة (مثل الحوسبة السّحابية) دوافع كبيرة للتَّطفُّل. لذا فإنّ على مهندسي الأمان أن يطوّروا تقنياتٍ مُبتكرة لمواجهة التهديدات التي تعترض أصول الشّبكات.

لقد ظهرت تقنيات جديدة متقدمة لإيجاد أنظمة أكثر فعالية لكشف التّطفّل باستخدام تقنيات تعلُّم الآلة وتقليل الحجم، وذلك لمساعدة مهندسي الأمان في التّوصُّل الى أنظمة عالية الفعالية في هذا المجال.

تقدم هذه الورقة مراجعةً نظاميّةً شاملةً لأنظمة كشف التّطفّل الأحدث التي استخدمت تقنياتٍ مراقبةً للتّصنيف في تعلّم الآلة وتقليل الحجم. وتوضّح هذه المراجعة كيف عملت مصنِّفات تعلّم الآلة وتقنيات تقليل الحجم على تطوير إنشاء أنظمة كشف التّطفّل؛ وذلك من خلال مقاييس التّقييم المعتمدة لمثل هذه الأنظمة.

وتجدر الإشارة الى أنّ النّقطة الأساسية في هذه الورقة تتمثّل في توفير أحدث المعرفة للباحثين المهتمين الجُدد في هذا المجال.