Expert Systems With Applications 198 (2022) 116822

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Review

A review of recent approaches on wrapper feature selection for intrusion detection

Javier Maldonado [a,*], María Cristina Riff [a,*], Bertrand Neveu [b]

[a] Departamento de Ingeniería Informática, Universidad Técnica Federico Santa María, Valparaíso, Chile
[b] LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Vallée, France

ARTICLE INFO

Keywords:
Intrusion detection
Wrapper feature selection
Literature review

ABSTRACT

In this paper, we present a review of recent advances in wrapper feature selection techniques for attack detection and classification, applied in intrusion detection area. Due to the quantity of published papers in this area, it is difficult to ascertain the level of current research in wrapper feature selection techniques. Moreover, due to the wide variety of techniques and datasets, is difficult to identify relevant characteristics among them, regard it architecture, performance, advantages and issues. The reported results frequently are shown in heterogeneous way, as there are several metrics to measure the classification quality. From our review, we propose a classification taxonomy of the wrapper feature selection techniques in intrusion detection area, considering design, rationale, technical characteristics and common evaluation metrics. Also we consider a description of the common metrics and a brief discussion about the attack scenarios reported in this review. At the end of this work, we show the coverage of existing research, open challenges and new directions.

1. Introduction

Nowadays, the intrusion detection problem is a hot topic in the cybersecurity and artificial intelligence community, due to the convergence of the everyday tasks on the IT infrastructures (Cisco Systems, 2018; Saloky & Šeminský, 2017), the increasing number and complexity of attacks (Navarro, Deruyver and Parrend, 2018; Tounsi & Rais, 2018) and the emerging technologies which are target of the new attackers as well (Roman, Lopez, & Mambo, 2018; Stellios, Kotzanikolaou, Psarakis, Alcaraz, & Lopez, 2018). Intrusion detection is a major research problem in network security and implies a mayor issue to IT world operations. An automated tool to inspect the networks and/or hosts activity are known as Intrusion Detection Systems (IDSs), which analyze information from the networks, trying to identify suspicious behaviors, also called attacks or intrusions, in order to detect, as soon as possible, actions that can affect the integrity, availability and confidentiality of an IT asset (Khraisat, Gondal, Vamplew, & Kamruzzaman, 2019).

To perform this detection the IDS, that is often inside or operate with the main firewall, read the raw data from the network and characterizes it into a flows using a set of features, which varies from the scenario point of view. Those flows are the representation of the interactions between the clients and the protected resources or servers. A large amount of those flows are analyzed by the IDS in a short time. Thus, an optimal subset of features allow to improve the IDS operations, in term of speed, accuracy, simplicity and response time. In Fig. 1 we show a general schema of an IDS in a network environment.

In our review, we observe that the experimental datasets report a wide variety and number of features. The major difficult in the intrusion detection is that the intrusion attempts are nonlinear, following an unpredictable behavior on the network traffic. The process of selecting the key features to obtain an effective IDS, is a crucial task in information security (Elhag, Fernández, Bawakid, Alshomrani, & Herrera, 2015; Kayacik, Zincir-Heywood, & Heywood, 2005)

In this paper, we present an overview of the research on wrapper feature selection, applied to intrusion detection systems (IDS). Many research work have been proposed in the literature, as shown in various surveys, covering different applications, such evolutionary computation approaches (Sen, 2015), describing briefly the IDS applications and point out the future directions and research opportunities, and Xue, Zhang, Browne, and Yao (2016) pointing out the benefits and applications of evolutionary algorithms in feature selection. Other focused in feature selection approaches by remark the difference between bio-inspired and non bio-inspired techniques (Balasaraswathi, Sugumaran, & Hamid, 2017), pointing out the structure, techniques and metrics used in each found approach. A survey focused in unsupervised applications (Nisioti, Mylonas, Yoo, & Katos, 2018), showing general classification, of the surveyed approaches, structure and metrics. A categorization focused surveys in machine learning techniques,

* Corresponding authors.
E-mail addresses: javier.maldonado@usm.cl (J. Maldonado), maria-cristina.riff@inf.utfsm.cl (M.C. Riff), bertrand.neveu@enpc.fr (B. Neveu).

https://doi.org/10.1016/j.eswa.2022.116822
Received 20 February 2021; Received in revised form 24 October 2021; Accepted 2 March 2022
Available online 18 March 2022

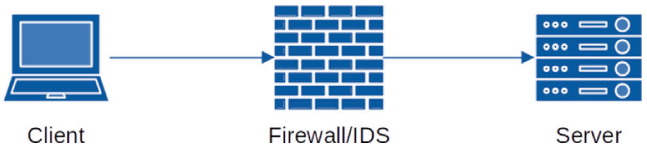

Fig. 1. IDS general schema.

Table 1
Key aspects of this review compared with others in the literature.

Reference	FS	Metrics	Taxonomy	Technical	Theoretical
Sen (2015)	F,W			X	
Xue et al. (2016)	F,E,W				X
Balasaraswathi et al. (2017)	F,W	X			X
Nisioti et al. (2018)	F,W	X		X	
Kumar et al. (2018)				X	X
Hindy et al. (2018)		X	TS	X	
Liu and Lang (2019)		X	TS	X	
Mishra et al. (2019)		X	AS	X	X
Resende and Drummond (2018a)	F,W	X		X	X
da Costa et al. (2019)				X	X
Bouzoubaa et al. (2021)	W	X		X	
Venkatesh and Anuradha (2019)	F,E,W	X			X
This work	W	X	WS	X	X

such (Kumar, Raju, & Vardhan, 2018), which classify the approaches regard the processing type, feature handling process and the used type of technique. In Hindy et al. (2018), which aim to provide a taxonomy and survey of intrusion detection system capabilities and threats; a specific machine learning and deep learning survey shown in Liu and Lang (2019), showing a taxonomy focused in the data sources and detection methods. One work focused on the taxonomy of the attacks shown in Mishra, Varadharajan, Tupakula, and Pilli (2019). A random forest specific survey is presented in Resende and Drummond (2018a), pointing out the advantages of these techniques in this particular unbalanced problem. One survey oriented to the internet of things (IoT) applications is shown in da Costa, Papa, Lisboa, Munoz, and de Albuquerque (2019) and one specific in denial of service and distributed denial of service attack types in Bouzoubaa, Taher, and Nsiri (2021). A review presented in Venkatesh and Anuradha (2019) focuses the feature selection benefits to handle large amount of information in the big data era.

The named surveys analyze the attack detection problem from different points of view. As far we know, there is not an up-to-date survey specifically focused on wrapper feature selection applied to the attack detection, which is our research area of interest and the scope of this review. We compare the considered key aspects of this work with surveys mentioned before, and note that none of them cover features selection classification methods, such filter, embedded or wrapper, considering aspects like evaluation metrics, describe common evaluation datasets, also if a is proposed a taxonomy and which aspects are considered on it. Also is the description of the technical aspects of the attacks and specific theoretical fundamentals of the problem are described. We resume the performed surveys comparison in the Table 1, each column represent the aforementioned key aspects of our review, where is an *X* in those aspects present in the corresponding reference. The values corresponding to column features selection (FS): (F) Filter, (E) Embedded and (W) Wrapper, in column Taxonomy: (TS) Technical Specific, which takes a technical aspects of the IDS as a criteria to categorize the approaches; (AS) Attack Specific, considers the tackled attack types to group the research work; or (WS) Wrapper Specific, that focus the categorization for wrapper features selection approaches specifically.

We observe differences of the surveyed approaches. Thus, we consider the characteristics shown in Table 1 in the approaches analysis and lead us to propose a taxonomy, to detail the main differences in the wrapper design in the observed research works. In our revision, we consider and describe the binary and multiclass attack classification problem, as a differential characteristic in the approaches design, due to each of them are evaluated using different metrics, according of the binary or multiclass abstraction of the observed scenarios, also briefly described in this work.

Thus, our motivation to perform this review, is to provide a structured and organized view of the wrapper feature selection applications on the attack detection problem, covering from the theoretical concepts, such problem characterization from the binary and multiclass classification, to technical aspects like attack types, techniques overview, evaluation metrics. Also, we propose a novel taxonomy that incorporates a wide view of the current and future techniques. We highlight challenges and advantages of the reviewed techniques. Our aim, is to show an unified vision of the existing wrapper, which are disseminated in many conferences and journals from different research domains. Furthermore, we have noticed that the experimental evaluations, metrics and datasets are not standardized, thus many proposed wrapper techniques are neither comparable nor reproducible. From our analysis of the experimental scenarios, we propose guidelines to allow the reproducibility in this research area. The main contributions of our work are:

- An updated review of the most recent proposed techniques for wrapper feature selection in algorithms that detect and classify attacks, in the field of cybersecurity.
- A classification taxonomy of wrapper feature selection techniques for attack detection and classification, considering characteristics as design, structure and application.
- An in-depth analysis of the experimental scenarios used in the literature for evaluating wrapper approaches.

The rest of the paper is organized as follows: we begin describing the intrusion detection problem in Section 2, focusing our discussion in the classification task. We remark the different point of view to define the attacks classification problem and later present, in Section 2.2, are described the most common metrics used to evaluate the attack classification quality. We finish this section analyzing the special difficulties related to this problem and remarking the role of the features selection methods to improve the classifiers performance. We briefly describe the supervised feature selection techniques applied in the context of IDS in Section 3, and we explain in detail the different wrapper approaches that belong to this class of techniques in Section 4. We propose a taxonomy in Section 4.1, as a result of our review. We analyze and discuss the technical characteristics of the surveyed approaches and compare them, in terms of experimental scenarios, metrics, wrapper techniques. We discuss about the advantages, issues and open challenges in Section 5. Finally, we present conclusions and future work in Section 6.

2. The intrusion detection problem

The Internet is now extensively used to make every day transactions, secure data transfers and digital certifications. These transactions rely on secure infrastructures in order to provide a reliable service. However, these infrastructures are perfect targets for attackers, that try to get access and compromise services in order to make profit (financial or technological) (Cisco Systems, 2018). In this context, the Intrusion Detection Systems (IDS) represent one of the essential building blocks in cybersecurity. The goal of an IDS is to mitigate unauthorized access to a given infrastructure by detecting, classifying and alerting intrusion attempts, in order to maintain availability, integrity and confidentiality (Stallings, Brown, Bauer, & Bhattacharjee, 2012). This is a difficult task, as it requires analyzing a large amount of real time data to detect potentially dangerous patterns that represent attacks. Moreover, the wide diversity of attacks makes detection even harder, due to the complexity of selecting key features to improve the detection effectiveness (Kayacik et al., 2005). Moreover, the fact that the attacks

are usually not uniformly distributed over time, increases the difficulty of detection.

The intrusion detection process performs the discrimination among a large amount of observed data (packets), which belong to different communications or interactions between users and/or devices. All the packets that belong to an interaction, have associated a set of data (source, destination, service, time window, among many others). In IDS, a preliminary process perform the values registration of the features that describe the interactions, a called characterization process, which generates the description of data flows, the basis of a behavior analysis.

2.1. Attacks in cybersecurity

First, a vulnerability in the field of cybersecurity, is defined as a weakness of an asset, in hardware or a piece of software, that can be exploited by an attacker or threat, with the objective of gain access to a valuable information or to cause damage to the enterprise. This is the origin of an attack (International Organization for Standardization, 2018).

In this context, we briefly define an *attack*, as any unauthorized activity that compromises the integrity, confidentiality and availability of an IT asset (Stallings et al., 2012). There is a wide variety of these activities, each of them pursuing a particular goal or to achieve a one step of a complex attack, thus an attack scenario could be composed by one or more actions with a common objective.

Most of the computer attacks fit on the following types. We present some of those attacks with a brief description (Khraisat et al., 2019; Mishra et al., 2019):

- *Denial of service (DoS):* the attacker objective is to leave a service unavailable by overloading the service requests capacity. The usual behavior is to perform a request storm in an very short time, with the objective of leave it temporally unavailable. A newer version with several origins is distribute denial of service (DDoS), which is more difficult to control due to its multiple sources structure.
- *Probing:* this kind of attack do not cause any damage on the objective. Its goal is to discover relevant information, for instance, about the network topology, services, configurations or any information that could reveal possible exploitable security issues of the system. Is a previous step of any further attack.
- *Remote-to-Local:* the attacker tries to gain unauthorized access to the local system from a remote location, taking advantage of a previous discovered security weakness.
- *User-to-Root (U2R):* or known as privilege escalation, once the attacker have access to a system, it will try to get administrative privileges (or root privileges), in order to take total control of the system. Is one of the most dangerous attacks, because the attacker compromise the entire system, probably, unnoticed.
- *Buffer overflow:* causes a memory violation, in which the program or routine end writing outside of it memory scope, with unpredictable results or leaving the attacker in a user shell, gaining unauthorized access to the system.
- *Worm:* a malicious program with the ability to reproduce itself into the system or network, with the objective of filter information or leave an unauthorized entrance to the attacker. The most it can reproduce, the more system infects and more chances to a successful attack it have.
- *Packet Flood:* is a type of DoS. The attacker flood the network with malformed packets, with the intention of consume all the infrastructure resources in the less possible time. Those packets are processed incorrectly by the hosts and network devices. The idea is to consume all the communication resources an cause a communication shortage in a wider area of the infrastructure.
- *Physical attack:* the attacker gain physical access to the IT resource and perform damage to it hardware or software, in a way that could be impossible in a remote attack. The attacker could leave a malicious device connected to the resource or its environment, with the intention to gain benefit or cause further damage to the infrastructure.
- *Password attack:* the attacker tries to guess a resource password using a dictionary, which contains several combinations of password that will try against the system, with the intention of gain unnoticed access to the objective. This kind of attacks can be detected by a several login errors reported in the system log.
- *Information gathering:* tries to obtain information by the exploration of the reachable infrastructure resources, in order to collect signals of services or devices with security flaws, and thus design an accurate attack strategy.

We have defined what is an attack and vulnerability, and described the most relevant attacks found in the literature. Nevertheless, as the technology advances, the attacks become more complex in its structure thus are harder to detect. Moreover, a vulnerability is the beginning of an attack, this is the attacker chance to take advantage of it and exploit the IT weakness to perform it malicious behavior. A vulnerability is detected if the attacker is able to acquire enough information of the objective, this is why the probing is considered an attack, it work as a first step in almost any attack. Much of them base its behavior on a sequence of simpler actions to build a complex and possible new attack scenario, also known as a zero-day attack, which are a never seen and new malicious behavior. As a starting point to detect such behavior, the research community base it efforts in build more accurate techniques to detect those steps in order to rebuild the full attack scenario and learn about this new behavior.

2.2. Attack classification problem

We identify two types of attack classification problems: a binary and a multi-class problem. In the case of binary attack classification, the goal is to distinguish between two classes of packets, an attack or a normal one. It implies that the main goal is to discriminate between these two kind of classes and not to identify which type of attack it is. When the problem is modeled as a multi-class attack classification, the goal is to identify a particular attack on a flow of packets that may contains different kind of attacks and normal ones. Formally, the attack classification problem can be formulated as:

Definition 1. Attack Classification Problem: Given a training set $\{(x_i, y_i)\}_{i=1,\dots,N}$ of N instances consisting of a packet $x_i \in X$ and a class label $y_i \in \{1, \dots, C\}$, with C as the number of classes and a given set F of k flow features $\{f_1, f_2, \dots, f_k\}$, the problem of learning a classification function $y : X- > \{1, \dots, C\}$ from the features and training set is called an Attack Classification Problem (ACP)

Remark 1. In the rest of the paper, when labels y_i have just two values Normal or Attack we address it as a Binary Attack Classification Problem (bACP) and when labels y_i involve more than one kind of attack and Normal values, we call it as a Multi-class Attack Classification Problem (mcACP)

There is a wide variety of metrics to measure the quality of a classification technique, some of them are oriented to the multiclass discrimination and other to the binary one (Milenkoski, Vieira, Kounev, Avritzer, & Payne, 2015; Pendleton, Garcia-Lebron, Cho, & Xu, 2016). In the following sections we show how to measure the classification success for bACP and mcACP.

Table 2
Example of binary-class confusion matrix.

		Prediction	
		Attack	Normal
Label	Attack	TP	FN
	Normal	FP	TN

Table 3
Example of multiclass confusion matrix.

		Predicted		
		A	B	C
Label	A	TP_A	e_{AB}	e_{AC}
	B	e_{BA}	TP_B	e_{BC}
	C	e_{CA}	e_{BC}	TP_C

2.2.1. Classification performance metrics for bACP

The Table 2 shows an example of a binary confusion matrix. In this context, the positive class is attack and the negative one is normal. Note that all kinds of attacks belong to the same attack class, thus, there is no difference among them.

Where true positives (*TP*) is the number of attack events correctly classified, false negatives (*FN*) is the number of attack events classified as normal, false positives (*FP*) is the normal events classified as attacks and true negative (*TN*) is the normal traffic correctly classified. The most popular classification success measures used in bACP are computed from the information in the associated confusion-matrix as Pendleton et al. (2016) and Tharwat (2018):

1. True positive rate (TPR):
   This measure is also known as detection rate (DR), sensitivity, hit rate or recall. It computes the proportion of correctly classified attacks over all the attack classified events. The goal is to maximize TPR:
   $$TPR = \frac{TP}{TP + FN} \tag{1}$$
2. Accuracy:
   The *Accuracy* metric defines the correct classification proportion (TP and TN) of all observed instances. Accuracy is computed as follows:
   $$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{2}$$
   High values of accuracy are preferred. Note that this metric could be biased to a bigger class, thus do not reflects well a poor classification quality of a small class.
3. False positive rate (FPR):
   Also known as fall-out or false alarm rate (FAR), this metric is oriented to the binary classification. It is calculated as follows:
   $$FPR = \frac{FP}{FP + TN} \tag{3}$$
   Then *FPR* is the proportion of misclassified normal traffic over all data. A lower value (close to 0) corresponds to a better classification of the normal class.
4. False negative rate (FNR):
   Also known as miss rate, oriented to the binary classification, is the proportion of attack samples classified as normal, and is computed as follows:
   $$FNR = \frac{FN}{FN + TP} \tag{4}$$
5. Precision:
   Is known as the proportion of correctly attacks classified over all attack predictions. A higher value denotes better classification quality. Is computed as follows:
   $$Precision = \frac{TP}{FP + TP} \tag{5}$$
6. Specificity:
   Also known as true negative rate (TNR) or inverse recall. In binary classification, is the ratio of the normal events correctly classified over all normal events. A higher value is suitable with a better classification quality. The computation of specificity is performed as follows:
   $$Specificity = \frac{TN}{TN + FP} \tag{6}$$
7. F1-score:
   Is the harmonic mean of precision and recall. A higher value of F1-score indicates a better classification performance. This metric is sensitive to the data distribution, that is the unbalanced data. Note that it does not consider the correct classified normal events (TN). Is computed as follows:
   $$F1score = \frac{2TP}{2TP + FP + FN} \tag{7}$$

2.2.2. Classification performance metrics for mcACP

In Table 3 we show a multiclass confusion matrix. Rows represent the known data labels and the columns the predicted values. The diagonal numbers represent the correct predicted values, or true positives (TP) of a class $TP_{\{A,B,C\}}$, and the off-diagonal elements correspond to a wrong classification, e_{EP}, with E and $P \in \{A, B, C\}$, $E \neq P$. Note that most of the following metrics are extensions of the binary classification metrics, its makes a reduction of binary classification on each calculated metric of a class.

In the following, we refer to TP_i as the number of elements correctly classified as attack class i, FN_i are those assigned to a wrong class, FP_i is the false positive prediction for a class i and M is the number of classes in the classification, then $2 \leq i \leq M$. Note that are the class-oriented extension of the bACP definitions. The most popular measures for mcACP use the information in the confusion matrix as Grandini, Bagli, and Visani (2020) and Tharwat (2018):

1. Recall:
   This measure is also known as sensitivity, hit rate, class oriented detection rate or true positive, is defined as the proportion of attacks class i correctly classified:
   $$Recall_i = \frac{TP_i}{TP_i + FN_i}, \ \forall\ i = 1 \ldots M \tag{8}$$
   From Table 3, TP_A in the row A, correspond to the correctly classified elements of the class A, in contrast with the rest of the elements in the row, e_{AB} and e_{AC}, that are known as class A false negatives. A higher values of $Recall_i$ $\forall i$ are better.
2. Precision:
   It is also known as positive predictive value (PPV), is defined as the ratio of correctly classified events in class i among all events predicted as class i:
   $$Precision_i = \frac{TP_i}{TP_i + FP_i}, \ \forall\ i = 1 \ldots M \tag{9}$$
   For instance in the column A, as shown in Table 3, the diagonal elements represent the true positives of the A class, the rest of the column items off-diagonal, represent false positives predictions. The higher precision is achieve, the lower number of false positives are in the classification. Thus, a higher values of $Precision_i$ $\forall i$ are suitable with a better classification quality.
3. Accuracy:
   An extension of this binary metric to the multiclass computation can be found in the literature, by calculating per-class *TP* and *FN* taking each class as a binary classification problem. The multiclass extension of accuracy is computed as follows:
   $$Accuracy = \frac{\sum_{i=1}^{M} TP_i + TN}{\sum_{i=1}^{M} TP_i + \sum_{i=1}^{C} FN_i + TN + FP} \tag{10}$$

Note that *FP* and *TN* are values related to the normal class. Note that this metric is sensitive to the unbalance data, biased to the bigger classes. Higher values of accuracy are preferred.

4. False negative rate (FNR):
This metric computes the proportion of the wrong classified events of class i. Referencing the Table 3, takes the row information of the corresponding class. This metric is computed as follows:

$$FNR_i = \frac{FN_i}{FN_i + TP_i} \tag{11}$$

5. False positive rate (FPR):
Also known as False alarm rate (FAR), this metric refers to the wrong predicted events of the class i over all the predictions of the class i. Considering the Table 3, takes the column information of the corresponding class. This metric is computed as follows:

$$FPR_i = \frac{FP_i}{FP_i + TP_i} \tag{12}$$

6. F1-score:
In the multiclass case of this metric, classes are considered, it can be computed in two ways: Macro F1-score and Micro-F1-score.
In the Macro F1-score case, a Macro Average Precision (MAvP) and Macro Average Recall (MAvR) are computed as:

$$MAvP = \frac{\sum_{i=1}^{M} Precision_i}{M} \tag{13}$$

$$MAvR = \frac{\sum_{i=1}^{M} Recall_i}{M} \tag{14}$$

thus the Macro F1-score is computed as follows:

$$Macro\ F1score = 2 * \frac{MAvP * MAvR}{MAvP^{-1} + MAvR^{-1}} \tag{15}$$

In Macro F1-score metric, all classes have the same importance, thus this metric reflects average performance of the classifier on the classification of all classes.
In the other hand, the Micro F1-score is computed from the Micro Average Precision (mAvP) and Micro Average Recall (mAvR), which are computed as follows:

$$mAvP = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M} TotalColumn_i} \tag{16}$$

$$mAvR = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M} TotalRow_i} \tag{17}$$

thus Micro F1-score is computed as follows:

$$Micro\ F1score = \frac{\sum_{i=1}^{M} TP_i}{Total\ of\ events} \tag{18}$$

Which is equal to multiclass accuracy. This measure gives more importance to bigger classes.

2.3. Remarks about classification task in IDS

Classification in IDS is a very hard task given its natural unbalanced flow of data. There is more normal than attack traffic in the data. Nowadays, it is important not only to discriminate between an attack and a normal traffic, it is also required to identify which kind of attack is, in order to take specific actions that allow to reduce its consequences. Moreover, new attacks are always in progress, thus the historical data could be useless for new scenarios. On the other hand, given the increasing sizes of available data and the need of use quality information, techniques for features selection are very important to help classifiers. When a classifier uses a reduced and relevant set of features, it is able to find a more interpretable model and can also dramatically improves its performance and readability (Nazir & Khan, 2019). The techniques for feature selection can be divided in supervised (Gan, Wen, Yu, Zheng, & Lei, 2020), unsupervised (Solorio-Fernández, Carrasco-Ochoa, & Martínez-Trinidad, 2020) and semisupervised (Ashfaq, Wang, Huang, Abbas, & He, 2017). Wrapper methods belong to the supervised feature selection techniques, thus in the following section we briefly describe those techniques.

3. Supervised feature selection techniques for intrusion detection

As we mentioned before, the IDS work is based on the observation of a set of features. However, not all features are relevant to perform a good classification. The problem is to select the most relevant features in order to obtain a good quality discrimination. This process is known as feature selection (Cai, Luo, Wang, & Yang, 2018). Formally, the problem of feature selection can be formulated as:

Definition 2. Given an Attack Classification Problem, F is a set of k flow features $\{f_1, f_2, \ldots, f_k\}$, and a quality classification measure Q, the problem of finding a subset F^* of F that allows to optimize Q, i.e. $Q(F^*)$ better than $Q(F)$, is called the Feature Selection Problem.

We can distinguish four classes of supervised feature selection techniques (Venkatesh & Anuradha, 2019):

(A) Filter feature selection, approaches that first select the most meaningful features and then performs the classification, using these selected (or filtered) features.
(B) Embedded feature selection, the features selection and classification tasks are embedded in a global algorithm, and these two parts are not separable.
(C) Interactive feature selection, which are those approaches that require an expert intervention to help the classification performance.
(D) Wrapper feature selection, the features selection and classification are two specific parts of an optimization global algorithm, that calls these two parts in an iterative loop process until a criteria is satisfied.

In the following, we give a brief description of each of these classes and we pay special attention in wrapper ones, which are detailed in Section 4.

3.1. Filter feature selection

In this group, we consider approaches that first filter all features during a preprocessing, and then use the selected subset of features for the classification. The filter features approaches (Guyon & Elisseeff, 2003), are oriented to keep the features with higher variance as a preprocessing step of the prediction, following the idea that those features with high variance have more relevant information to a classification model. Generally, these approaches have low execution time but may choose redundant variables. In Fig. 2 we show a generic scheme of these approaches.

The Fig. 2 shows that, given a Full Set of Features, a process to select the best subset of features is performed. It is done by a filtering process that obtains the best subset of features which will be used by the classification algorithm. Finally a classification model is created.

Different approaches based on correlation exist, which uses the information of a class given by a related set of features, known as correlation based feature selection (CFS) (Hajisalem & Babaie, 2018; Su et al., 2019; Yılmaz Gündüz & Çeter, 2018; Zhou, Cheng, Jiang, & Dai, 2020). From these filtering selection techniques, some use feature variance, in order to detect patterns that allow them to discriminate a particular behavior, such as Multivariate Mutual Information based Feature Selection (MVMIFS) (Mohammadi, Desai, & Karimipour, 2018), Principal Component Analysis (PCA) (Salo, Nassif, & Essex, 2019), Chi-Square test (Divyasree & Sherly, 2018; Jiang & Xu, 2019).

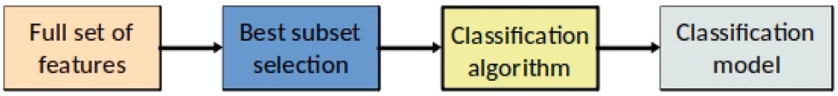

Fig. 2. Filter method scheme.

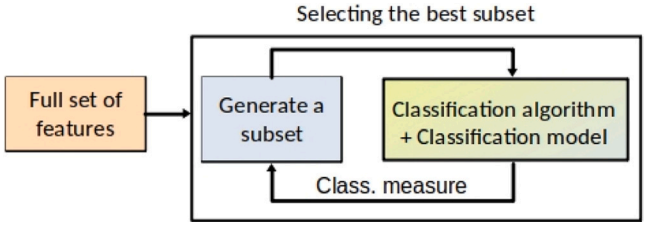

Fig. 3. Embedded method scheme.

Others techniques of filter features selection use probability of a prediction technique (Ahmim, Derdour, & Ferrag, 2018) and cluster models analysis (Jiang et al., 2018).

Most of these techniques use well known classification tree algorithms (Abdullah, Alshannaq, Balamash, & Almabdy, 2018; Ahmim et al., 2018; Jiang & Xu, 2019; Yılmaz Gündüz & Çeter, 2018) given their performance and simplicity. Other works use Support Vector Machines (SVM) (Hosseini Bamakan, Wang, Yingjie, & Shi, 2016; Mohammadi et al., 2018; Salo et al., 2019) and Core Vector Machine (CVM) (Divyasree & Sherly, 2018), taking the advantage of their high-dimensional space separation abilities.

In other approaches the authors propose different subset of features according to each attack class, in an attempt to discriminate attacks more accurately (Abdullah et al., 2018), a specific attack (Jiang et al., 2018) or detect zero day attacks (Hamed, Dara, & Kremer, 2018).

3.2. Embedded feature selection

These approaches do not separate the feature selection process from the classification technique. The feature selection is performed during the classification process, thus it is included into the feature selection as a functionality extension or a component of the algorithm itself (Khorram, 2020).

In Fig. 3, is shown a components interaction schema of the embedded feature selection approaches, based on its generic structure.

As the classification technique is integrated with the feature selection procedure, the features subset needs to be evaluated during the construction, thus, the evaluation metric also is included in the process as internal information in the algorithm, shown in Fig. 3.

In our revision, we found some works related to the embedded techniques. Regularization based techniques (Bao, Muñoz González, & Lupu, 2018; Kozik & Chora, 2019) are focused on mitigate the overfitting through the minimization of the intrinsic difficulties of the data, such as data unbalance nature or misinformation added by the attackers. Some techniques, like Multilayer Perceptron (MLP) (Wang, Lu, & Qin, 2020), searches for identifying non-linear classes focused on the information provided by the features. Other techniques use the descriptive information of the features and the relation among them, like correlation coefficient (Mohd Yusof, Mokhtar, Zain, & Maple, 2018) and recursive features addition (Hamed et al., 2018; Nancy et al., 2020). Another approach takes the Bayesian statistic fundamentals, which compare a prior event class from a supposed data distribution with the real one, as a Bayesian estimator (Jiang, Li, De Rijke, Yao, & Chen, 2019). Finally, considering the search power of metaheuristic based techniques, a named gravitational search algorithm, is presented in Zhu, He, Wang, Zeng, and Yang (2019), which model the features selection problem as a combinatorial optimization one.

3.3. Interactive feature selection

In this group, we present approaches that need an expert human intervention at some stage of its execution, in order to improve the classification performance.

An adaptive approach is presented in Resende and Drummond (2018b), which is based on anomaly detection with a genetic algorithm to create a behavior profile baseline, in order to detect variances over a normal network traffic using two anomaly-based techniques. The approach in Arnaldo and Lam (2019), proposes a framework to detect anomalies which produces a features subset based on principal component analysis (PCA) technique. Then the detected anomalies or outliers, are validated by a security staff. Other approaches tackle the multi-step attack detection problem with staff expertise (Navarro, Legrand, Deruyver and Parrend, 2018; Navarro-Lara, Deruyver, & Parrendz, 2017), correlating events and refining the detection capabilities using the human experience.

In the following section we focus our attention on wrapper feature selection methods applied to IDS.

4. Wrapper feature selection for IDS

We present here a review of approaches that use wrapper feature selection technique to help classifiers. These approaches (Kohavi & John, 1997), use two clearly separated techniques, one that select features separated from a classification technique. This kind of approaches usually models the feature selection problem as a combinatorial search one. In Fig. 4 is shown the wrapper approach schema.

A feature subset is evaluated by a Classification Algorithm, which returns its classification measure. This procedure is performed until a stopping criterion is reached. Finally, the best found subset is used to build the final model.

An IDS uses a wrapper algorithm to automate and improve the update of its knowledge base, which is often based on rules that needs to be updated periodically by the security staff or the IDS manufacturer. So, a well designed wrapper algorithm integrated in the IDS, allows a faster rule update process, which is an advantage in the attack response that needs to be faster as possible. We show the principal components of an IDS that use a wrapper technique in Fig. 5.

Analyzing the feature selection methods, we found approaches that use the features correlation information to separate all classes (Aljawarneh, Aldwairi, & Yassein, 2016; Anwer, Farouk, & Abdel-Hamid, 2018; Kamarudin, Maple, Watson, & Safa, 2017; Latha, 2018; Mohammadi, Mirvaziri, Ghazizadeh-Ahsaee, & Karimipour, 2019; Wahba, ElSalamouny, & ElTaweel, 2015) and features correlation in each class (Liu, Wang, Tao, & Cai, 2015), custom generated trees (Botes, Leenen, & De La Harpe, 2017), cluster (Soheily-Khah, Marteau, & Bechet, 2018) and statistical models (Zhang, Qu, & Deng, 2018). Their goal is to select those features that help the classification algorithm to be more performant.

Recently, there is an increasing number of algorithms based on bio-inspired techniques proposed in the literature, like evolutionary and genetic algorithms, some of them tackle the feature selection problem as a combinatorial optimization one, trying to find those features with the best performance (Almasoudy, Al-yaseen, & Idrees, 2020; Maldonado & Riff, 2019; Maldonado, Riff, & Montero, 2019; Vijayanand, Devaraj, & Kannapiran, 2018; Xue, Jia, Zhao, & Pang, 2018), others optimizes the features and the classificator parameters at the same time (Gauthama Raman, Somu, Kirthivasan, Liscano, &

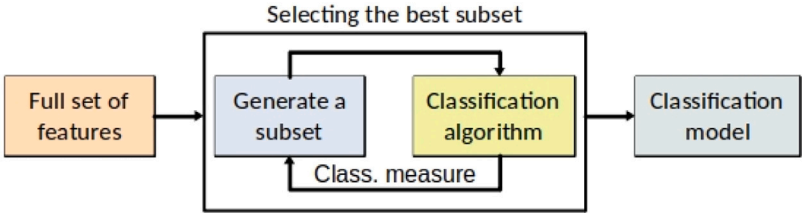

Fig. 4. Wrapper method scheme.

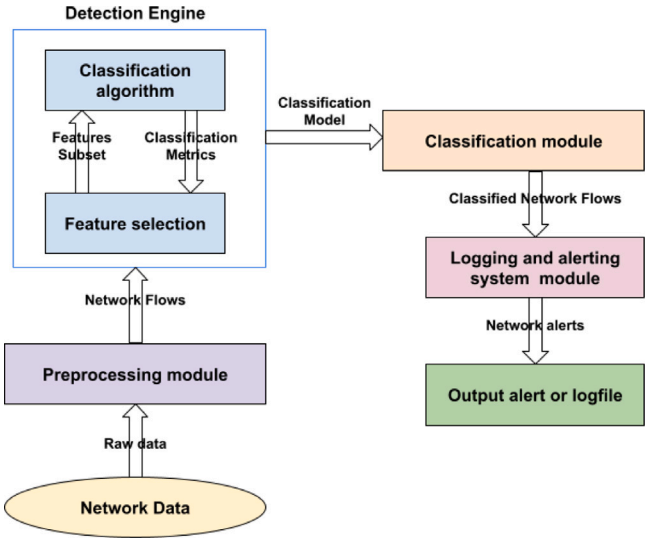

Fig. 5. A general scheme of an IDS with wrapper technique.

Shankar Sriram, 2017; Gharaee, Fekri, & Hosseinvand, 2018) or taking a multiobjective approach of feature selection (Khammassi & Krichen, 2020). The swarm based approaches, take advantage of its exploration abilities in complex, multidimensional and very big attack search spaces, like Artificial Bee Colony (ABC) (Mazini, Shirazi, & Mahdavi, 2018), Particle Swarm Optimization (PSO) (Li, Guo, Wu, & Li, 2018), Firefly algorithm (FFA) (Selvakumar & Muneeswaran, 2019) and a combination of Whale Optimization Algorithm (WOA) and a genetic algorithm (Vijayanand & Devaraj, 2020).

Other techniques, that uses custom or combination of classifiers, are ant tree miner classifier (Gauthama Raman et al., 2017), which uses artificial ants to generate a tree classifier for each class; a combination of different classification techniques using kernel-based feature selection (Botes et al., 2017); a custom genetic algorithm that perform feature selection and parameter optimization (Xue et al., 2018); differential evolutionary feature selection (Mazini et al., 2018) and a kernel based features selection (Zhang et al., 2018).

4.1. Wrapper feature selection for IDS - taxonomy

In our review, we note similarities on the wrapper feature selection approaches applied in the attack detection area.

We propose a taxonomy, shown in Fig. 6, which is centered in the wrapper techniques and consider three main characteristics: detection focus, technique used and architecture.

- *Detection focus of the ACP (Fo-ACP):* As we mention before, the classification can be focused on the binary ACP (bACP), where the detection can be focused on *Normal traffic* (N), tackling the normal classification as a negative class. The bACP focused on normal traffic, detect abnormal behavior taking the advantage of the amount of normal data available, which is generally greater than the attack data. The bACP approaches focused on *Attack events* (A), center their effort detecting malicious traffic no matter their specific class, all events are grouped in a non-normal traffic (or positive class of the binary problem). Learn from the attack data is a big challenge, due to its constant change and few available data. In the multiclass ACP (mcACP), we consider three classes of focus. *Attack detailed* (AD), those approaches that keep track of the attack classes and are able to notify which attack behavior is. These approaches are complex, due to the amount of detailed data to process. The *Class focused* (AD-C) approaches are able to differentiate one particular kind of attack, in most of these approaches the detection is performed with the technique one-over-all focusing in the information about a specific behavior, by extracting part of the data or keeping track of this specific behavior. The *All classes* (AD-A) approach can detect various types of attack present in the data, discriminating the normal traffic and focusing it effort only in the attack behaviors, allowing to detect more than one attack using the same model.
- *Techniques (Tec):* As wrapper feature selection involves a collaboration between selection and classification tasks, we identify which kind of technique perform the wrapper feature selection and which one the classification (i.e. machine learning (ML), statistical (St) or metaheuristic (MH)). Note that, as there is a wide diversity of techniques, some of them are oriented to search near-optimal solution in a high complex space (i.e. metaheuristics and evolutionary approaches), or a classification tasks (such the machine learning approaches). There are several ways to integrate techniques.
- *Architecture (Arch):* It specifies how the classifier uses the data. It can be *Hierarchical* (H), where the traffic is detected by decomposing the data in smaller sections, which allows to simplify the problem instance temporally, in order to achieve more accurate results focusing on a temporally reduced data frame. Other detection approaches take the classification technique considering all information and try to differentiate among classes, called *General* (OL-G), or one in particular, *Specialized* (OL-S). This can be focused on one or more classes or create several classification instances, each of them specialized in one or more classes of attacks.

In Table 4 we present the revised work according to our taxonomy. In the first column named Ref, we show the reference that is briefly described in the column Key Aspects. In the column named C. Alg. is show the classification algorithms implemented, in Dataset column we name the experimental datasets used. Finally, in the columns Fo-ACP, Tech and Arch, we show the corresponding characteristics regard the proposed taxonomy.

5. Discussion

In this section, we discuss the wrapper feature selection techniques found in this review we present an extensive experimental analysis considering different scenarios using various existing datasets.

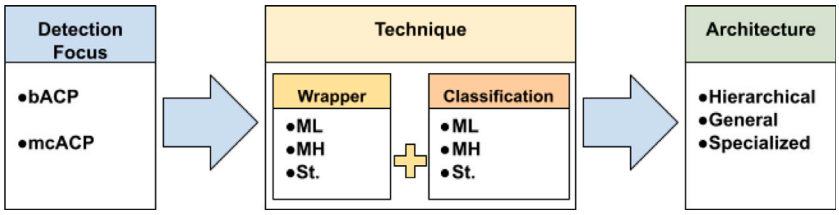

Fig. 6. Wrapper selection features taxonomy.

5.1. Approaches analysis

In the following, we analyze different approaches presented in the Table 4. As a result, we present a summary of the approaches grouped by dataset and organized in tables as follows: In the Ref. column, we show the work reference, the Wrapper and Class columns show the wrapper and classification technique reported by the authors, or the best one in case of the wrapper were tested with more than one classification algorithm. If some approaches are shown more than once in a given table, it means that the authors show results in bACP and mcACP terms for this specific scenario. In the columns Tech and Fo-ACP, we use the proposed taxonomy (see Section 4.1), to assign a category according to the observed combination of techniques and the type of problem abstraction, respectively. Finally, in the columns bACP Metrics and mcACP Metrics, are shown the reported metrics grouped by the type of problem. Note that we sort the tables by the number of approaches of each found dataset. For a comprehensive overview of the available datasets for testing attack detection techniques, refer to Ring, Wunderlich, Scheuring, Landes, and Hotho (2019) regard the network IDS, or those related to host IDS in Bridges, Glass-Vanderlan, Iannacone, Vincent, and Chen (2019).

We note that each of the reported datasets denotes a different attack scenario, with marked differences, like amount of features, events and classes. All of them are unbalanced, as the normal/attack network traffic is. Moreover, some events of attack classes represent less than the 1% of the traffic, and normal one often represent close the 50% or more. Most of the approaches perform a data preprocessing in different ways, normalization, feature prefiltering, repeated subset elimination, all of them aiming to improve the detection and classification quality, starting from the data source management.

5.1.1. Approaches using NSL-KDD dataset

The NSL-KDD dataset was created by the Faculty of Computer Science, University of New Brunswick, Canada. It provides attack and normal scenario having different features. This artificial dataset is well known and widely used in the evaluation of intrusion detection algorithms. It contain five traffic classes: DoS (Denial of Service), U2R (User to Root), R2L (Remote to Local), Probe and Normal traffic, all of them labeled and described by 41 features values on each sample (Tavallaee, Bagheri, Lu, & Ghorbani, 2009).

Some of the surveyed works takes the complete dataset to perform its experiments and other uses a subset of its data, this depends of the researchers design and focus. For sake of comparability, we present the approaches that perform its test with the full set of data, that is the full scenario information. Nevertheless, we note a relevant characteristics of the surveyed approaches, most of them uses a bio-inspired approaches, but slightly different among them, due to its highly customization structure according with the researcher goal design and the approach focus of study.

In the Table 5 we group the observed approaches that solve the bACP using the full NSL-KDD dataset with the corresponding evaluated metrics.

In the work of Sarvari et al. (2020), they perform the feature selection using a modified cuckoo fuzzy algorithm, which incorporate a mutation operator that is able to change various genes simultaneously, in order to improve the search space exploration. To mitigate the class overlap, the result are postprocessed with Fuzzy C-means, which improve the class discrimination. Finally, the parameters of the classification algorithm are tuned with a multiverse optimization algorithm. This approach combines various techniques to help the classification algorithm.

A pigeon inspired optimizer is presented in Alazzam et al. (2020), which perform the feature selection taking advantage of the exploration ability of the algorithm. The authors perform a discretization of the continuous operation mode of the algorithm, searching to minimize the amount of features and maximize the metric values.

In Almasoudy et al. (2020) is proposed a differential evolution approach with an extreme learning machine, which is able to manage high dimensional and non-linear data, and a variable length representation, aiming to reduce the construction time and improve the classification accuracy.

A custom design approach is the combination of known techniques, in order to take advantages or tackle the combined techniques issues, as is shown in Samadi Bonab et al. (2020), that combine fruitfly algorithm (FFA) with ant lion optimizer (ALO), in order to take advantage of the FFA high convergence and the ALO breadth exploration in the high dimension space solution.

In the Table 6 we group the observed approaches that solve the mcACP using the NSL-KDD dataset with the corresponding reported metrics.

This kind of approaches need to discriminate among different kind of attacks, which is a harder task as some of the behaviors are more complex that just discriminate among normal and not normal ones, also it must indicate what kind of attack is. We found multiobjective approaches as feature selection method, such (Golrang et al., 2020), which is a combination of NSGA-II and ANN to perform a elimination of redundant subset of features, in order to help the diversity of the population and avoid the premature convergence. As the objective functions they minimize the classification error and the size of the features subset. A GA approach that perform the feature selection is shown in Liu and Shi (2022), which evaluation function, as an important behavior modifier of the algorithm, is a weighted composition of accuracy, FPR and F1. The resulting feature subset is evaluated using RF. In Wei et al. (2020) another multiobjective approach is show, they optimize the Jaccar's coefficient of each detected class, in order to perform the find the subset of features that maximize the classification quality of these classes. Note that is a modified standard and well known algorithm, adjusted to the context problem, it incorporate a logistic regression to refine the subset of features. Note that (Almasoudy et al., 2020) appears again solving th mcACP with th NSL-KDD, so this approach is able to solve both problems, which demonstrates it adaptability to different abstraction of the considered scenario.

We also note a combination of techniques in mcACP approaches, such (Hosseini & Zade, 2020) than combines a genetic algorithm with a SVM to perform the feature selection. Moreover, each operator of the GA is an individual algorithm and the representation is the algorithm parameters to be optimized, then the classification is made using a ANN. The same author in Hosseini (2020) proposed a combination of GA with a logistic regression, in order to calculate the feature correlation, refine the GA feature selection and, in consequence, focus the

Table 4
Revised wrapper attack detection works according to the proposed taxonomy.

Ref.	Key aspects	C. Alg.	Dataset	Fo-ACP	Tech	Arch.
Botes et al. (2017)	An ant tree miner (ATM) classifier, which creates decision trees using ant colony optimization based algorithm.	DT	NSL-KDD	mcACP, AD-A	MH-ML	OL-S
Gauthama Raman et al. (2017)	An adaptive technique using Hypergraph based Genetic Algorithm (HG - GA) for feature selection and parameter optimization of the classification technique.	SVM	NSL-KDD	bACP, A	MH-ML	OL-G
Kamarudin et al. (2017)	HFS, which has a filter and wrapper parts using a genetic search algorithm, to reduce the amount of features.	RF	NSL-KDD, UNSW-NB15	mcACP, AD	MH-ML	OL-G
Anwer et al. (2018)	A framework that uses a wrapper feature selection based on J48 and NB, helped with another filter technique based in feature correlation.	J48, Naïve Bayes	UNSW-NB15	bACP, A	ML-ML	OL-G
Latha (2018)	A custom feature selection called high pertinent feature selection mechanism (HPFSM), which work associated with IG and CFS.	ANN, SVM	DARPA	bACP, A	ML-ML	OL-G
Vijayanand et al. (2018)	A genetic algorithm based approach to select relevant features for each category of attack instead. Was tested on a mesh wireless network topology.	SVM	ADFA-LD, CICIDS2017	mcACP, AD-A	MH-ML	OL-S
Xue et al. (2018)	A self adaptive differential evolution (SADE) algorithm that generates a candidate solutions or feature subset combinations	KNN	KDD99	bACP, A	MH-ML	OL-G
Gharaee et al. (2018)	A genetic algorithm that optimize the feature vector and SVM classifiers.	SVM	UNSW-NB15, KDD99	mcACP, AD-A	MH-ML	OL-G
Mazini et al. (2018)	An artificial bee colony (ABC) algorithm to select relevant features that maximizes the detection quality of a classificator method.	AdaBoost	NSL-KDD, ISCXIDS2012	mcACP, AD-A	MH-ML	OL-G
Li et al. (2018)	An hybrid method based on RF and PSO to select an optimal subset of features and optimize the algorithm parameters, in order to improve the RF classification quality.	RF	KDD99	bACP, A	MH-ML	OL-G
Selvakumar and Muneeswaran (2019)	A firefly algorithm to select a subset of features to train two classifiers and evaluate the classification quality.	C4.5, Bayesian Network	KDD99	mcACP, AD-A	MH-ML	OL-G
Kamarudin, Maple, and Watson (2019)	An hybrid approach that combines CFS with a three search algorithm: best-first, greedy stepwise and genetic algorithm. The generated subsets of features are evaluated using RF as a wrapper classifier.	RF	KDD99, DARPA	bACP, A	MH-ML	OL-G
Mohammadi et al. (2019)	IDS feature selection and clustering algorithm, using wrapper method based on cuttlefish algorithm (CFA) and helped with linear correlation coefficient.	ID3	KDD99	bACP, A	MH-ML	OL-S
Maldonado et al. (2019)	An custom evolutionary algorithm to select automatically key features from a dataset to build a decision tree based classifier.	C4.5	NSL-KDD	mcACP, AD-A	MH-ML	OL-G
Maldonado and Riff (2019)	Evaluate different metric configurations based on two well known metrics of a custom evolutionary algorithm, to select optimal subset of features from a dataset.	RF	NSL-KDD	mcACP, AD-A	MH-ML	OL-G
Almasoudy et al. (2020)	A differential evolution algorithm used to select an optimal subset of features, from a given dataset, to train an extreme learning classifier.	ELM	NSL-KDD	mcACP, AD-A	MH-ML	OL-G
Khammassi and Krichen (2020)	Multi-objective detection approach, based on the NSGA-II algorithm, for feature selection to train a binomial and multinomial logistic regression classification algorithm.	C4.5, RF, NBT	NSL-KDD, UNSW-NB15, CICIDS2017	mcACP,AD-A	MH-ML	OL-S
Vijayanand and Devaraj (2020)	A modified whale optimization algorithm (WOA), which crossover and mutation operators are modified to overcome the premature convergence issue of the algorithm.	SVM	CICIDS2017, ADFA-LD	mcACP, AD-A	MH-ML	OL-G
Polat and Polat (2020)	A combination of different classification algorithms with the Sequential Forward Floating Selection algorithm as a wrapper technique to detect a DDOS attack in a SDN infrastructure.	NB, ANN, KNN	Proprietary	mcACP, AD-C	ML-ML	OL-S
Kurniabudi et al. (2020)	A PSO approach to select a subset of optimal features to be evaluated by J48 implementation in Weka.	C4.5	CICIDS2017	mcACP, AD-A	MH-ML	OL-G
Kasongo and Sun (2020)	Implements a Wrapper Based Feature Extraction Unit (WFEU), a custom method based on extra-tree classifiers.	DNN	UNSW-NB15	mcACP, AD-A	ML-ML	OL-G
Sarvari, Mohd Sani, Mohd Hanapi, and Abdullah (2020)	An anomaly detection method based on the Mutation Cuckoo Fuzzy (MCF) for feature selection. Each solution is also evaluated with Fuzzy C Means (FCM) clustering.	ENN	NSL-KDD	bACP, A	MH-ML	OL-G
Solani and Jadav (2021)	Creates a subset of features using recursive feature elimination (RFE) and perform the evaluation with linear discriminant analysis as a classification technique.	LDA	UNSW-NB15	bACP, A	ML-ML	OL-G
Alazzam, Sharieh, and Sabri (2020)	Perform the feature selection using a Pigeon Inspired Optimizer (PIO). Propose the binarization of the continuous metaheuristic in order to solve discrete problems.	DT	NSL-KDD, UNSW-NB15, KDD99	bACP, A	MH-ML	OL-G

(*continued on next page*)

Table 4 (*continued*).

Ref.	Key aspects	C. Alg.	Dataset	Fo-ACP	Tech	Arch.
Umar, Zhanfang, and Liu (2020)	An hybrid approach based on BestFirst algorithm, which evaluates its features subsets with a different classification algorithms.	Random forest, Naive Bayes, KNN, SVM, ANN	UNSW-NB15	bACP, A	MH-ML	OL-G
Almomani (2020)	A C4.5 and SVM classifiers are tested with various feature selection algorithms: PSO, GWO, FFA and GA. A comparison of those techniques are performed.	C4.5, SVM	UNSW-NB15	bACP, A	MH-ML	OL-G
Samadi Bonab, Ghaffari, Soleimanian Gharehchopogh, and Alemi (2020)	Proposes an hybridization with the Fruit fly algorithm (FFA) and ant lion optimizer (ALO), then the generated subsets are evaluated with four different classifiers.	SVM, KNN, NB, DT	NSL-KDD, UNSW-NB15, KDD99	bACP, A	MH-ML	OL-G
Sarıkaya and Kılıç (2020)	A custom wrapper feature selection design is proposed in a hierarchical model, in order to focus its efforts in a particular class on each iteration.	RF, KNN, NN	UNSW-NB15	mcACP, AD-A	MH-ML	H
Wei, Chen, Lin, Ji, and Chen (2020)	Proposes a multiobjective approach to evaluate different class of events in the tested dataset.	GHSOM	NSL-KDD, UNSW-NB15	mcACP, AD-A	MH-ML	OL-S
Golrang, Golrang, Yayilgan, and Elezaj (2020)	A multiobjective approach implemented with an hybridization with a modified NSGAII with integrated ANN to generate the best possible feature subsets.	RF	NSL-KDD, UNSW-NB15	mcACP, AD-A	MH-ML	OL-G
Shafiq, Tian, Bashir, Du, and Guizani (2020)	Combines accuracy and feature correlation as a wrapper method to select and remove features from a generated near optimal subset, oriented to detect bot attack in an IoT infrastructure.	C4.5, RF, DT, NB	Bot-IoT	mcACP, AD-A	ML-ML	OL-G
Kalaivani and Gopinath (2020)	Introduces an hybrid method composed by bacterial foraging optimization (BFO) and modified bee colony (MBC) algorithms, analyzed with three different classification techniques.	NN, ReNN, RNNs	KDD99	bACP, A	MH-ML	OL-G
Davahli, Shamsi, and Abaei (2020)	Presents a combination of a genetic algorithm and grey wolf optimizer as a wrapper approach (GABGWO) to select features in a wireless sensor network of an IoT infrastructure.	SVM	AWID	bACP, A	MH-ML	OL-S
Abbasi, Al-Sahaf, and Welch (2020)	A PSO based method to detect ransomware in two phases approach: data transformation (ranking features in subgroups with mutual information) and feature selection (using PSO).	RLR, RF, DT, SVM	Custom	mcACP, AD-A	MH-ML	OL-G
Hosseini and Zade (2020)	A combination of SVM with a genetic algorithm based on multi-parent crossover and multi-parent mutation. Additionally a PSO and HGS algorithms are used to improve the feature selection performance.	ANN	NSL-KDD	mcACP, AD-A	MH-ML	OL-G
Syarif, Afandi, and Astika Saputra (2020)	A CSA is proposed to select subset of features on three datasets. Its results are compared with PSO and GA.	DT	NSL-KDD, KDD99, Bot ISCXIDS2017	bACP, A	MH-ML	OL-S
Hosseini (2020)	This approach is divided in two stages: first stage uses genetic and logistic algorithms to find a correlated subset of features. The second stage, attack detection, the ANN algorithm is trained by particle optimization (PSO) and gravitational search (GS) algorithms to perform the attack classification.	ANN	NSL-KDD, KDD99	mcACP, AD-A	MH-ML	OL-G
Maldonado and Riff (2020)	Custom evolutionary algorithm which consider the feature importance information in the custom roulette feature selection process. The feature importance is provided by a RF algorithm.	RF	NSL-KDD	mcACP, AD-A	MH-ML	OL-G
Liu and Shi (2022)	Design a custom genetic algorithm with a special composed evaluation function, combining accuracy, false positive rate and F1 metrics, all of them with different metrics weights.	RF	NSL-KDD, UNSW-NB15	mcACP, AD-A	MH-ML	OL-G
Nazir and Khan (2021)	Perform the feature selection with a Taboo Search algorithm and its fitness function is inspired in a neural network.	RF	UNSW-NB15	mcACP, AD-A	MH-ML	OL-G

Table 5
Approaches solving bACP with full NSL-KDD dataset.

Ref.	Wrapper	Class.	TPR	Acc	FPR
Sarvari et al. (2020)	MCF	ANN	97.25	98.81	0.02
Alazzam et al. (2020)	PIO	DT	86.60	88.30	8.80
Almasoudy et al. (2020)	EA	ELM	82.12	87.53	
Samadi Bonab et al. (2020)	FFA	ALO	99.24	99.31	

search in more relevant subsets. The classification is made by an ANN trained and optimized by a PSO and gravitational search algorithms. This is a combination of techniques, each of them oriented to a specific task.

In Botes et al. (2017) is proposed an algorithm based on ant tree miner to build a decision tree. They claim the lack of reliability, comparability and reproducibility, thus hey argue the approach description is clear enough to replicate the experiment. The swarm based approaches, take advantage of its exploration abilities in complex, multidimensional and very big attack search spaces, like Artificial Bee Colony (ABC) (Mazini et al., 2018), which is dedicated to process large amount of data to find a minimal set of performant features.

We found another approaches that perform its experiments with a subset of the NSL-KDD, thus its results are not comparable, due to the differences in the amount of events per class. Nevertheless, there are a relevant information that we observe from these approaches. The approach (Khammassi & Krichen, 2020) performs an stratified sample of the data, considering the size and proportion of the classes, in order to mitigate the unbalance nature of data and improve the experimental performance. They use a multiobjective NSGA-II, which minimize the feature subset size and maximize the accuracy. To evaluate the optimal subsets of features, they perform the classification with RF, C4.5 and Naïve Bayes tree. In Syarif et al. (2020) the authors perform the feature selection with a cuckoo search algorithm, which explore the search

Table 6
Approaches solving mcACP with full NSL-KDD dataset.

Ref.	Wrapper	Class.	Recall	Prec	Acc	FPR	F1-score
Hosseini and Zade (2020)	GA	ANN	97.50	95.20			0.96
Hosseini (2020)	GA	ANN	94.40	94.50			
Golrang et al. (2020)	NSGAII-ANN	RF	99.40	99.40	99.40	6.00	
Liu and Shi (2022)	GA	RF			96.12	2.91	
Mazini et al. (2018)	ABC	ADABoost	99.61		98.90	0.02	
Botes et al. (2017)	ATM	DT	57.05		64.85		0.73
Almasoudy et al. (2020)	EA	ELM	41.54	67.51	80.15		
Wei et al. (2020)	MOIA	GHSOM			99.47		

Table 7
Approaches solving bACP with UNSW-NB15 full dataset.

Ref.	Wrapper	Class.	TPR	Acc	FPR	FNR	Prec	Spec	F1-score
Anwer et al. (2018)	C4.5	C4.5		86.60					
Almomani (2020)	GA	C4.5	96.70	86.87	21.16	3.30	78.89	78.83	0.86
Alazzam et al. (2020)	PIO	DT	89.40	91.70	3.40				0.90
Umar et al. (2020)	BF	RF	97.95	86.41	27.73				
Kasongo and Sun (2020)	WFEU	DNN		85.48					
Almomani (2020)	GA	SVM	96.97	86.38	22.27	3.03	78.08	77.73	0.86
Samadi Bonab et al. (2020)	FFA	ALO	93.46	99.12					

space in a parallel and independent way. They perform the experiment with the training part of the dataset. An specific behavior detection is performed in Kamarudin et al. (2017), which focus its detection and classification efforts in the HTTP attacks, thus they build a subset of data containing this related events. The feature selection is performed with a RF algorithm, taking advantage of it random sampling and feature importance abilities. An evolutionary approach (Gauthama Raman et al., 2017) includes the parameters of the SVM classification algorithm in a mix of real and binary features representation. This, in order to minimize the subset of features and tuning the SVM parameters at the same time. To perform the validation with this scenario, they generate a random subset without replacement from the training and testing portions of the data, this generates a variety of individuals or models with a different levels of behavior specialization.

5.1.2. Approaches using UNSW-NB15 dataset

The UNSW-NB15 dataset was created by the University of New South Wales for the evaluation of network IDS's. It has an update collection of legitimate and anomalous behaviors, is a mix of real scenario data and synthetic network events, which provides a good approach to current real world scenarios, with the goal of include the most novel available behaviors at its time (Moustafa & Slay, 2015).

Is the second dataset on appearances in our review. In Table 7 we show the reported approaches using the UNSW-NB15 full dataset in bACP.

Most of the shown approaches reports its results as the bACP classification problem, in terms of accuracy and most of them in TPR. The bio-inspired feature selection strategies are the most popular, just three approaches report non-bioinspired strategies (Anwer et al., 2018; Kasongo & Sun, 2020). In the case of Anwer et al. (2018) is considered only HTTP related events. Use the C4.5 algorithm as the feature selection procedure, due to its fast management ability of large amount of data, and Naïve Bayes due to its simplicity in the predictions, note that both algorithms are most common as classification than as wrapper feature selection strategies. The approaches (Almomani, 2020; Umar et al., 2020) reports the higher FPR values. The first approach argue improve the base line value with a less selected features (only 19 of 41), its a simple algorithm that takes a best first feature to add in the feature subset and build the optimal one. The second one perform the feature selection by a GA, which produces a subset suitable set of rules that can be combined in order to provide a different classification goals. So, different group of features lead to wide different results, no matter the metric or the classification strategy. They aím to create a specialized classificators of different behaviors. In the case of Kasongo and Sun (2020), it build the subset of features vector using an extra tree algorithm, to test the strategy with network and wireless traffic scenarios, which are able to perform the classification from the bACP and mcACP perspective. So, this approach can manage is flexible enough to manage the ACP in both abstraction of this scenario. An approach presented in Samadi Bonab et al. (2020), takes advantage of the high convergence rate of the fruitfly algorithm (FFA) and improves it exploration ability with the ant lion optimizer (ALO), in order to avoid the local optima stagnation problem of the FFA. In the case of Alazzam et al. (2020), we found that the approach was also tested with the NSL-KDD scenario, which demonstrate its adaptability to another different and more complex scenario.

In the Table 8 we show the approaches in mcACP that report results using the full set of UNSW-NB15.

Regard to Gharaee et al. (2018), evaluate the classification quality of the SVM classification, with a compose metric that considers the minimization of the amount of features and FPR, and the TPR maximization, thus considers three aspects of the feature selection problem, which guides the GA to optimize the solution considering more classification information. Also is possible to generate a separate behavior profile using a hierarchical approach, such (Sarıkaya & Kılıç, 2020), which generate a set of feature subsets specialized in a particular behavior, so this approach is able to focus in some particular event by the separation of the data in sub-scenarios. Thus this approach is able to provide a profiling based features subset. In Nazir and Khan (2021) the feature selection is performed using a Taboo Search algorithm considering the feature correlation information, in order to improve the space search procedure. The evaluation function is inspired in the cost function of the neural network model. The classification is performed with the RF algorithm. Thus, the incorporation of this information provided by the preprocessing stage of this approach, help the feature selection process. Regard (Kasongo & Sun, 2020), note that also report results in this scenario with bACP results, thus we can observe an adaptive ability to both abstraction of this scenario.

Note that (Golrang et al., 2020; Liu & Shi, 2022; Wei et al., 2020), also validate its approaches with the full NSL-KDD dataset, showing an adaptive ability in different scenarios.

Table 8
Approaches solving mcACP with UNSW-NB15 full dataset.

Ref.	Wrapper	Class.	Recall	Prec	Acc	FPR
Golrang et al. (2020)	NSGAII-ANN	RF	94.80	94.80	94.80	6.00
Nazir and Khan (2021)	TS	RF			83.20	3.70
Liu and Shi (2022)	GA	RF			92.06	1.60
Gharaee et al. (2018)	GA	SVM	96.83		90.08	0.39
Kasongo and Sun (2020)	WFEU	DNN			74.78	
Wei et al. (2020)	MOIA	GHSOM	97.05	95.11	94.60	
Sarıkaya and Kılıç (2020)	Custom	KNN	97.00		81.00	

Table 9
Approaches solving bACP with a subset of CICIDS2017 dataset.

Ref.	Wrapper	Class	Acc
Khammassi and Krichen (2020)	GA	DT	99.39
Syarif et al. (2020)	CSA	CFS	99.98

Table 10
Approaches solving mcACP with a subset of CICIDS2017 dataset.

Ref.	Wrapper	Class	Recall	Acc	FPR
Vijayanand et al. (2018)	GA	SVM		99.39	0.32
Vijayanand and Devaraj (2020)	WOA	SVM	96.77		0.76
Khammassi and Krichen (2020)	GA	C4.5	85.32	95.15	0.35
Kurniabudi et al. (2020)	PSO	C4.5		99.88	

5.1.3. Approaches using CICIDS2017 dataset

The CICIDS2017 dataset, was created by the Canadian Institute of Cybersecurity, University of New Brunswick, Canada (Sharafaldin, Habibi Lashkari, & Ghorbani, 2018). It defines an scenario with 78 features to characterize each type of activity (flow) in the dataset, which is comprised by eight files with the traffic monitored on a real network laboratory for a week. This dataset is oriented to build scientific reference data for experiments in the cybersecurity area, and contains a set of events that builds complex network attack scenarios, which are composed of single or several flows (normal and/or attacks), distributed on these files. All the reported approaches uses a subset of the dataset, this is due to its size and complexity.

In Table 9 we show the approaches tackling the bACP with the CICIDS2017 scenario. Note that these approaches uses a subset of the dataset, which makes not fair comparable the reported results, as all of them are performed with different size and configuration of the data.

The approach (Khammassi & Krichen, 2020), appears twice in this scenario, due to they test the stratified subsets of data with two different classification algorithms, DT to solve the bACP instance and C4.5 to solve a mcACP instance. Note that this approach also appears using NSL-KDD and UNSW-NB15 datasets, all of them with a stratified subset of data. Thus, is able to manage both abstractions of the tested scenarios, showing adaptability to different situations. The approach (Syarif et al., 2020), report only accuracy metric in this scenario, also we describe it in Section 5.1.1 and show results also in with the scenario KDD99, thus the authors perform validation of its approach with different scenarios, demonstrating the approach adaptability to different situations.

In Table 10 we show the approaches tackling the mcACP with the CICIDS2017 scenario. Note that these approaches uses a subset of the dataset, which makes not fair comparable the reported results, as all of them are performed with different size and configuration of the data.

In Vijayanand et al. (2018) is performed the selective train of several SVM classificators, each of them specialized in a particular behavior, thus this approach is able to generate a profile based on the feature subset of teach SVM. Its generate a several simpler and accurate models instead of create a single complex one. Note that this approach is performant in all classes, as the accuracy and FPR reports, thus this is a promising design. A modified whale optimization algorithm is presented in Vijayanand and Devaraj (2020), which includes a crossover, selection and mutation operators from a genetic algorithm, in order to mitigate the stagnation issue of the original algorithm and preserve the best individual features during its process, thus this operator allow to perform a breadth exploration of the search space, finding better solutions and avoiding fall in a local optima. The approach (Kurniabudi et al., 2020), regroup the dataset in seven types of attacks and normal class, which reduces the original 16 classes. This helps the algorithm to classify, in a more general way, the attacks but lose details about the features involved in the specific ones. This approach is able to generalize about the attack profiles. Note that (Khammassi & Krichen, 2020) appears again in this abstraction, as we mention before in the bACP description of this scenario.

5.1.4. Approaches using DARPA dataset

In Table 11 we show the approaches using the DARPA dataset (Lippmann et al., 2000), which are one of the most well known and older of the experimental dataset for IDS applications. Is criticized for the injected artificial traffic and the amount of data redundancy. Created by the MIT Lincoln Laboratory within an emulated network scenario for a five weeks, it characterizes attacks like rootkits, portscan, DoS and buffer overflow, in network flows, which are built from the packet based data extracted from the emulated environment. Note that specificity and F1 were removed from the Table 11, as none of the reported approaches report results in those metrics.

We found just three approaches using the DARPA dataset, all of them solving the bACP whit this scenario. One of them (Latha, 2018) proposes an approach based on a information gain wrapper feature selection. It analyzes the impact of each feature incorporation to the subset under construction, aiming to obtain a minimal subset of features with a performant metrics. It perform a classification with ANN and SVM. We note that the ANN obtain better results than SVM on all the reported metrics by this approach. Both classification techniques are able to separate non-linear high dimensional spaces, but the wrapper algorithm is able to found a subset of features more suitable to ANN classification algorithm. The approach (Kamarudin et al., 2019), select a subset of the related events to a specific host with the major amount of attacks in this scenario. They perform a feature filter with correlation feature selection prior to the feature selection with the GA, in order to improve model construction time of the approach, by the reduction of the GA search space. This approach takes advantage of the intrinsic correlation information of the data.

5.1.5. Approaches using ADFA dataset

The ADFA-LD dataset was developed to benchmark host based IDS (HIDS). This scenario is composed by host activities in Windows and Linux environments, using the latest publicly available exploits, in its time. It is composed by system call traces, instead of network flows, that consist in tree main groups: training, testing normal and testing attack. As it design is oriented to the anomaly detection, there is no attack traces in the training set. The attack and normal traces share similarities, which become a challenge to create accurate approaches. The Windows specific dataset is oriented to recreate scenarios from zero-day attacks, stealth attacks, data exfiltration and DDoS (Creech & Hu, 2013).

In the Table 12 we show the approaches using a subset of the ADFA scenario.

Table 11
Approaches solving bACP with DARPA dataset.

Ref.	Wrapper	Class	Data	TPR	Acc	FPR	Prec
Latha (2018)	IG	ANN	Full	83.00	86.20	0.24	83.00
Latha (2018)	IG	SVM	Full	81.00	85.39	0.31	81.00
Kamarudin et al. (2019)	GA	RF	Subset	99.51	99.93	0.01	

Table 12
Approaches solving mcACP with a subset of ADFA dataset.

Ref.	Wrapper	Class	Recall	Prec	Acc	FNR	FPR
Vijayanand et al. (2018)	GA	SVM			96.95	15.30	1.25
Vijayanand and Devaraj (2020)	WOA	SVM	86.30	77.36			0.82

Table 13
Approaches solving bACP with other subset datasets.

Ref.	Wrapper	Class	Dataset	TPR	Acc	FPR	Prec	F1
Davahli et al. (2020)	GABGWO	SVM	AWID	99.30	99.09	0.68	96.31	0.97

Table 14
Approaches solving mcACP with other full datasets.

Ref.	Wrapper	Class	Dataset	Recall	Acc	FPR
Shafiq et al. (2020)	CorrACC	RF	Bot-IoT	92.30	99.39	0.26
Mazini et al. (2018)	ABC	ADABoost	ISCXIDS2012	86.00	83.00	5.00

We found two approaches oriented to detect attack events in a wireless mesh network in this scenario. Both of them tackle the mcACP and use different subsets of data, thus there is not possible perform a fair results comparison. Also, both are tested with a subset of CICIDS2017 dataset. Note that, despite that this dataset is oriented to HIDS, those works take a subset of events that reflects the activities in a wireless mesh environment, thus this works tackle the problem of network intrusion detection from the host point of view, a relevant difference among the other works. Moreover, those approaches show its flexibility, as they are tested with different scenarios, aiming to the attack detection and classification with different points of view.

5.1.6. Approaches using others public dataset

Here is shown those datasets that we found only once in our survey. The AWID dataset (Kolias, Kambourakis, Stavrou, & Gritzalis, 2016) is oriented to attack detection in only wireless scenarios, a recent one called Bot-IoT dataset (Koroniotis, Moustafa, Sitnikova, & Turnbull, 2019), which is oriented to internet of things infrastructure scenario. The ISCX2012 (Shiravi, Shiravi, Tavallaee, & Ghorbani, 2012) was created by capturing traffic in an emulated network scenario over one week, is the predecessor of CICIDS2017.

In the Table 13 we show the unique approach that use the AWID dataset solving the bACP.

The shown approach in Davahli et al. (2020) is oriented to perform the bACP in a wireless environment. The authors argue that the wireless sensors deployed in a IoT network generates a large amount of data, that is not easy to process in a short time. They propose a wrapper feature selection with combination of a modified grey wolf optimizer, incorporating crossover and mutation operators from a genetic algorithm. This combination is made in order to avoid the local optima and improve the feature exploration. The features subset are evaluated with a SVM algorithm, which is able to discriminate non linear and high dimensional data.

In Table 14 we show the approaches that tackle the mcACP with other datasets.

The approach reported in Shafiq et al. (2020) is tested an IoT simulate scenario deployed in a home, considering the mcACP with 8 classes (including normal). They propose the wrapper strategy based on the mathematical analysis of the features correlation and perform the classification using the random forest algorithm. In this case the feature correlation is not a preprocessing step, is taken as the feature selection method, which is a uncommon way to build the feature subsets. In the approach (Mazini et al., 2018), the authors tackle the mcACP from the anomaly based point of view, with an artificial bee colony algorithm (ABC) and ADABoost as classification technique. They use the ISCXIDS2012 dataset, oriented to the attack detection and classification in a network environment, is the predecessor of the previously named CICIDS2017 dataset, also this approach appears in Section 5.1.1, using the NSL-KDD dataset, thus this approach show an adaptability to a complex scenarios.

Table 15
Approaches solving mcACP with full proprietary dataset.

Ref.	Wrapper	Class	Acc
Polat and Polat (2020)	SFFSA	KNN	98.30
Abbasi et al. (2020)	PSO	RF	54.68

5.1.7. Approaches using proprietary datasets

Finally, in the Table 15 we show the approaches using proprietary datasets, which are build for a specific scenario or from data that are not publicly available for any reason. All the approaches shown in the Table 15 report it results in accuracy terms.

The approach (Polat & Polat, 2020), is focused in the detection of DDoS attacks in a SDN network scenario, using a sequential forward floating feature selection, which is a greedy method that explores the combinatorial effect of the features and is able to determine the interaction among them, in order to find a best subset of features, which is evaluated using a KNN algorithm. In Abbasi et al. (2020) use a custom design dataset created from a resource of goodware and ransomware in Sgandurra, Muñoz González, Mohsen, and Lupu (2016). This scenario is described by 30,967 features grouped in seven categories. The most 400 important features of each group are selected, using mutual information as a previous step in the feature selection process. Then, the wrapper feature selection is performed with PSO and the resulting subsets of features are evaluated using RF. Note that this approach are able to handle a large amount of features.

5.2. Wrapper and classification analysis

We can distinguish two kind of techniques in our review, the bio-inspired and non bio-inspired approaches. In the following we describe

them, differentiating briefly the works and techniques that belong to each group.

The Bio-inspired approaches groups the major number of the reviewed works, all of them often combined well-known machine learning techniques as classification methods. The aim of these approaches is to take advantage of the exploration power of the wrapper techniques, which in most cases transform the feature selection into a combinatorial problem. Some of these approaches were tested with different classification methods, which denotes the adaptability and flexibility of those kind of techniques. In our review, we note a diversity of approaches, from the classical techniques, like genetic algorithms (Almomani, 2020; Gauthama Raman et al., 2017; Gharaee et al., 2018; Hosseini, 2020; Hosseini & Zade, 2020; Kamarudin et al., 2019, 2017; Khammassi & Krichen, 2020; Vijayanand et al., 2018), evolutionary algorithms (Almasoudy et al., 2020; Maldonado & Riff, 2019; Maldonado et al., 2019; Xue et al., 2018), artificial bee colony (Mazini et al., 2018) or particle swarm optimization (Abbasi et al., 2020; Kurniabudi et al., 2020; Li et al., 2018). Also other newer techniques such as firefly algorithm (Samadi Bonab et al., 2020; Selvakumar & Muneeswaran, 2019), cuttlefish algorithm (Mohammadi et al., 2019), pigeon inspired optimizer (Alazzam et al., 2020), cuckoo search algorithm (Syarif et al., 2020) or whale optimization (Vijayanand & Devaraj, 2020) shown promising results in this context. Moreover, due to the versatility and adaptability characteristics of the bio-inspired techniques, other combination or hybrid approaches are proposed, such modified genetic algorithm (Davahli et al., 2020), modified bee colony (Kalaivani & Gopinath, 2020) and mutation cuckoo fuzzy (Sarvari et al., 2020). Also we found works oriented to multiobjective approaches (Golrang et al., 2020; Wei et al., 2020), which models some problem characteristic as many objectives, showing another way to tackle the attack classification problem.

Concerning non bio-inspired approaches, most of them are a combination of techniques from the machine learning discipline, considering their abilities in managing high dimensional data in classification tasks. A combination of wrapper techniques are found, named correlation accuracy (Shafiq et al., 2020), sequential forward floating selection (Polat & Polat, 2020), naïve bayes (Anwer et al., 2018), information gain (Latha, 2018), recursive feature elimination (Solani & Jadav, 2021) and a tree-based approaches (Anwer et al., 2018; Kasongo & Sun, 2020).

In Table 16 we show the type of wrapper approach on the rows and classification techniques using this type of wrapper.

Concerning techniques oriented to perform attack classification, we found that the top classification technique of the reviewed work is support vector machines (SVM), Abbasi et al. (2020), Almomani (2020), Davahli et al. (2020), Gauthama Raman et al. (2017), Gharaee et al. (2018), Latha (2018), Polat and Polat (2020), Shafiq et al. (2020), Umar et al. (2020), Vijayanand and Devaraj (2020) and Vijayanand et al. (2018), aiming to separate high dimensional attack data and its ability to separate cases even if they are less than the number of features. In second place we find tree based approaches, taking advance of their performance, easy of use and interpretability, like ID3 (Mohammadi et al., 2019) and C4.5 (Almomani, 2020; Anwer et al., 2018; Khammassi & Krichen, 2020; Kurniabudi et al., 2020; Liu et al., 2015; Maldonado et al., 2019; Selvakumar & Muneeswaran, 2019; Shafiq et al., 2020) which considers categorical and/or discrete data and gives the ability to convert the resulting trees in if-then rules, which can be applied to a higher level detection method. Additionally, the well-known assembly of trees, Random Forest (RF) (Abbasi et al., 2020; Kamarudin et al., 2017; Khammassi & Krichen, 2020; Li et al., 2018; Liu & Shi, 2022; Maldonado & Riff, 2019; Nazir & Khan, 2021; Shafiq et al., 2020; Umar et al., 2020), which have a better unbalance data management than the other tree based algorithms.

Approaches based on artificial neurons classifiers, we found artificial neural networks (ANN) (Hosseini, 2020; Hosseini & Zade, 2020; Kalaivani & Gopinath, 2020; Latha, 2018; Polat & Polat, 2020; Sarvari et al., 2020; Umar et al., 2020), with an static structured perceptron net based, for creating a non-linear detection models in complex search spaces. Another approach includes linear structures of perceptrons classifier such recurrent neural networks (RNN) and a hierarchical one like recursive neural networks (RvNN) (Kalaivani & Gopinath, 2020). A supervised cluster based technique, K-Nearest Neighbors (KNN), is presented in Polat and Polat (2020) which tries to predict labels from a closest distance of a fixed number of attack or normal samples.

In Table 17 we show the experimental datasets related to the reviewed works. In the rows, is shown the classification techniques and in the columns the datasets. Note that here we do not differentiate the bACP from the mcACP approaches, just group the classification techniques vs. the datasets.

The datasets shown in the Table 17 characterizes a set of complex events coming from the network or host activity, that is why those datasets require several features to describe those events or interactions, thus a lot of information is needed to discriminate accurately among all the registered network flows.

As shown in Table 17, the tree-based classificators are the most popular classification technique in the review, followed by the SVM. Note that these approaches use the most popular datasets, UNSW-NB15 (Moustafa & Slay, 2015), NSL-KDD (Tavallaee et al., 2009), KDD99 (Tavallaee et al., 2009), CICIDS2017 (Sharafaldin et al., 2018). Despite of most of them are criticized for its age and represent an outdated cybersecurity scenarios, are still in use by the scientific community, due to the wide knowledge about its advantages and issues, thus they are a reliable start point to perform a sound test of incipient approaches. Also we found other datasets, such AWID (Kolias et al., 2016) oriented to attack detection in only wireless traffic, SCUT (Moore & Zuev, 2005) and UNIBS (Gringoli, Salgarelli, Dusi, Cascarano, Risso, et al., 2009), which are oriented to network traffic classification. A most recent one Bot-IoT dataset (Koroniotis et al., 2019), which is oriented to the internet of things infrastructure.

In the column Proprietary are shown those works that uses proprietary dataset such as Polat and Polat (2020) and Custom those that generate its own dataset from a combination or manipulation of existing one, as Abbasi et al. (2020) do.

5.3. Best bACP approaches

We show in Table 18 the best approaches found by dataset regard the bACP type. The selection was performed considering the reported results in true positive rate (TPR), accuracy (Acc), false positive rate (FPR), false negative rate (FNR), precision (Prec) and F1 terms. Those approaches that achieve best values over all with a given full dataset, are reported in this table. Ref. column shows the reference of the shown work, Dataset column, as named, is the dataset used to evaluate the approach and, the left of the table, are the reported results in its corresponding named columns.

Note that a common reporting metrics are TPR and Acc, they are the easiest to interpret and implement in the binary classification problems. So, they are able to indicate how the classifier method performs in term of a correct overall classification. The FPR metrics, in this case, reports the classification quality of the normal class.

Here, we describe the main approaches characteristics shown in the Table 18, in order to point out why those works are successful with its corresponding dataset.

The approaches evaluated using the UNSW-NB15 dataset in terms of TPR, the best value is achieved by Umar et al. (2020), it performs a preprocessing step that consist in eliminating redundant features, normalize numerical values to avoid classification bias and encoding the categorical values to make easier the classification data handle. The evaluation is performed with a tree-based classifier, taking advantage of its categorical and numerical features handle abilities and its low training and classification time. Note that there is an important effort in the preprocessing data step, in order to mitigate issues, such as

Table 16
Detailed wrapper vs. classification techniques.

Wrapper	Classification technique																			Total
	SVM	C4.5	RF	ANN	DT	KNN	NB	RF	NBTr	ALO	GHSOM	AdaB	ELM	ID3	LDA	DNN	BN	RvNN	RLR	RNN	
GA	6	4	7	3				2	3												25
PSO	1	1	2		1														1		6
FFA		1								3							1				5
BF	1		1	1		1	1														5
CorrACC	1	1	1				1														4
EA		1	1			1							1								4
SFFSA	1			1		1	1														4
PIO					3																3
CSA					3																3
Custom		2				1															3
MBC-BFO				1														1		1	3
ABC												2									2
NSGAII-ANN								2													2
Naive Bayes		1					1														2
MOIA											2										2
IG	1			1																	2
WOA	2																				2
C4.5		1					1														2
WP						1															1
WFEU																1					1
RFE															1						1
MCF				1																	1
GABGWO	1																				1
CFA														1							1
ATM					1																1
Total	14	12	12	8	8	5	5	4	3	3	2	2	1	1	1	1	1	1	1	1	84

Table 17
Classification techniques vs. datasets in wrapper feature selection approaches.

Class.	Dataset														Total
	UNSW-NB15	NSL-KDD	KDD99	CICIDS2017	Proprietary	Bot-IoT	Custom	DARPA	ADFA-LD	AWID	ISCX2012	ISCX2017	SCUT	UNIBS	
Random forest	6	1	2	1		1	1	1							17
SVM	3	1	1	2	1	1	1	1	2	1					14
C4.5	4	2	1	2		1							1	1	12
ANN	1	3	2		1			1							8
DT	1	3	2				1					1			8
Naive Bayes	3				1	1									5
KNN	2		1		2										5
ALO	1	1	1												3
NBTree	1	1		1											3
GHSOM	1	1													2
AdaBoost		1									1				2
ID3			1												1
LDA	1														1
ELM		1													1
DNN	1														1
Bayesian Network			1												1
RLR							1								1
RNN			1												1
RvNN			1												1
Total	25	18	14	6	5	4	4	3	2	1	1	1	1	1	84

Table 18
Best bACP approaches by dataset.

Ref.	Dataset	bACP metrics						
		Class.	TPR	Acc	FPR	FNR	Prec	Spec	F1
Umar et al. (2020)	UNSW-NB15	RF	97.95	86.41	27.73				
Samadi Bonab et al. (2020)	UNSW-NB15	DT	93.46	99.12				91.76	
Almomani (2020)	UNSW-NB15	SVM	96.97	86.38	22.27	3.03	78.08	77.73	0.86
Sarvari et al. (2020)	NSL-KDD	ANN	97.25	98.81	0.02				
Samadi Bonab et al. (2020)	KDD99	ANN	99.87	99.73				99.67	
Kamarudin et al. (2019)	DARPA	RF	99.51	99.93	0.01	0.49			

bias, underfitting or overfitting, caused by the dataset structure. The best Acc value is achieved by Samadi Bonab et al. (2020), it proposes a combination of two bio-inspired algorithm to perform the feature selection: the fruit fly algorithm (FrFA), a population based algorithm based on the behavior of the fruit fly in finding food, this algorithm is easy to implement, has few parameters and fast convergence, is suitable to perform optimization in high dimensional spaces. To overcome the FrFA to be trapped in a local optima or solutions with a

lower fitness, the ant lion optimizer (ALO) is implemented, which is also a population based algorithm and imitates the interaction of ants lion and ants in hunting. One of the advantages of the ALO, is its exploration power which avoid it to be trapped in a local optima, due to its elitism characteristic. Also the dataset is prepared by a feature mapping and normalization as a previous step of the experimentation. The classification achieves the best result with decision tree as a classifier and the UNSW-NB15 and KDD99 datasets, according to the Table 18. In Almomani (2020) we can observe the lowest results expressed in terms of FNR, this approach combines a filter technique (mutual information) with the wrapper approach that, at the same time, combines the given selected features by firefly algorithm (FFA), grey wolf optimizer (GWO), particle swarm optimization (PSO) and genetic algorithm (GA), in order to produce rules based on the selected features. The evaluation is performed with SVM and C4.5 algorithms. The dataset is prepared with the elimination of the attack category and normalizing data, aiming to make an homogeneous classification. There is a combination of techniques in this approach, which aim to improve the feature selection from different metrics and, in consequence, from different aspects of the problem, which makes this design more flexible to focus the algorithms in some particular behavior. Also this approach reports its results in terms of FNR, Prec, Spec and F1, which allows to analyze in detail the behavior of the classification results from the positive and negative class point of view.

In Sarvari et al. (2020), evaluated with NSL-KDD dataset, is shown the best values of TPR, Acc and FPR. It implements the modified Cuckoo Search Algorithm (CSA), called Mutation Cuckoo Fuzzy (MCF), which uses mutation to improve the details in the examination of the search space, in order to allow candidate solutions to escape from local optima. As classification algorithm, the Evolutionary Neural Network (ENN) is trained with the output features subset of MCF and the value of the solution is evaluated with the objective function and the Fuzzy C Means (FCM) clustering method to improve the results taking advantage of the solution overlapping and create the fuzzy membership search domain, including all possible compromise solutions, which improves its exploratory abilities. This approach aims to explore intensively the search, space creating clusters of solutions that could overlap among them, taking advantage of the non-linear separation of the problem and set the boundaries of the two classes with a fuzzy function.

With the DARPA dataset, we found (Kamarudin et al., 2019) as the best reported values. In this approach, the authors propose an hybrid method in three stages. In the first stage, the initial full set of features is filtered using a genetic algorithm with correlation based feature selection (CFS). Later, in the second stage, the features selection is performed with a wrapper genetic algorithm using a random forest as a classification algorithm. And, in the third stage, the generated RF model is evaluated with the KDD99 and DARPA datasets using ten fold cross validations. They explore the significance of the features in the first stage and refine the left features with a metaheuristic method, which helps to a RF classification technique to improve its performance. They perform a feature elimination using intrinsic information provided by the relationship among the features.

Despite the dataset or metrics, there are some relevant approaches characteristics. It seems a good practice to implement dataset preprocessing, such features normalization, calculate the correlation or importance among features or encoding them according to the approach implementation, in order to mitigate the intrinsic issues of this kind of data. Combining several techniques, could result in a complex non-easy to explain and hard to implement solution, thus this would be a trade off between complexity and efficiency. Regard the classification techniques, most of them aims to uses advantage of well known techniques from the machine learning discipline, preferring those that can handle high dimensional and unbalance data, two main characteristics of the shown datasets. Note that there is a preference to metrics that evaluates the correct classification, despite the size of the classes, such TPR and Accuracy, followed by FPR, which aims to measure the quality of the classification from the negative class error point of view.

5.4. Best mcACP approaches

We show in Table 19 the best works found by dataset regard the mcACP approaches. The selection was performed considering the reported results in recall, precision (Prec), accuracy (Acc), false positive rate (FPR) and F1-score (F1) terms. Ref. column shows the reference, Dataset shows the used dataset to evaluate the approach and, in the rest of the table, the reported results in its corresponding columns. Note that the shown approaches are selected from those that uses the full named dataset, thus the results can be compared.

The approaches shown in the Table 19 show results in terms of Recall, Precision and Accuracy, in spite of accuracy and recall are not the most suitable to multiclass classification on unbalanced data (Grandini et al., 2020; Tharwat, 2018), they are easy to implement and interpret.

Using UNSW-NB15 as experimental dataset, in Sarıkaya and Kılıç (2020) is presented a hierarchical multiclass approach focused in the classes with lower detection rates. In each stage of detection, the algorithm creates a specialized classifiers to detect different attack classes, those with lower detection rate are treated individually as bACP, in order to increase the classification quality and taking advance of the unbalance nature of the dataset. In this hierarchical model, in stage 1, a random forest classifier is trained to discriminate among normal and not normal traffic. In stage 2 other random forest classify the non normal traffic (or attacks) in group 1 (DoS, Exploit) and group 2 (other attack classes). Finally, in stage 3, two different random forest classifiers perform the detailed classification of the group 1 and 2 respectively. This approach focus the detection efforts in classes at each stage, in order to perform a detailed class detection and eliminate the noisy information to a specific class, but possibly required to another in the next stage. This approach takes advantage of the unbalance nature of the data and adapt the classification aiming to benefit the smaller classes.

For the NSL-KDD dataset, we found one approach that report a good results in terms of Rec, Prec, Acc and FPR (Golrang et al., 2020). It proposes a multi-objective approach based on the NSGA-II algorithm and a artificial neural network (ANN) that run simultaneously to perform the feature selection and tries to reduce the features redundancy, in order to improve the classification quality in the selected features. The subset evaluation is made using a random forest algorithm. The dataset is preprocessed, mapping the nominal features into integer values and all features values are normalized in a [0–1] range.

In the approach (Hosseini & Zade, 2020), the results are reporting in Rec, Prec and F1. This approach perform the feature selection using a combination of an evolutionary algorithm, SVM and ANN. The evolutionary algorithm and SVM perform the feature selection, later, for improving its classification performance, a combination of a hybrid gravitational search (HGS) and a particle swarm optimization (PSO) is used to train an artificial neural network (ANN) as the classification algorithm. The dataset is preprocessed, converting all data in numerical values and, later, normalizing its values into [0–1] range, in order to avoid the values bias and improve the results of the ANN classifier. This approach achieve good balance between per class precision and recall, as shown by F1 metric values, thus the classification quality achieved, although is not the best of the Table 19, is good enough to give a high value of this metric. This combination achieves good performance in the minority (thus harder) classes, which is an improvement of the classification technique through feature selection.

An approach with KDD99 dataset is presented in Hosseini (2020), which performs the classification in two stages. The feature selection stage, uses a genetic algorithm and a logistic regression algorithm to find a correlated subset of features. In the attack detection phase, the ANN is trained by particle swarm optimization (PSO) and gravitational search (GS) algorithms, which aims to optimize the weights of the ANN. The dataset is preprocessed, converting the nominal features into a numeric values and then normalize all numerical values in the dataset, this is in order to avoid classification bias by the dataset scale

Table 19
Best mcACP approaches by dataset.

Ref.	Dataset	mcACP metrics					
		Class.	Recall	Prec	Acc	FPR	F1
Sarıkaya and Kılıç (2020)	UNSW-NB15	DT	97.00		81.00		
Golrang et al. (2020)	NSL-KDD	RF	99.40	99.40	99.40	6.00	
Hosseini and Zade (2020)	NSL-KDD	ANN	97.50	95.20			0.96
Hosseini (2020)	KDD99	ANN	96.83	97.20			
Kurniabudi et al. (2020)	CICIDS2017	C4.5			99.88		

values. The training performed by the combination of the PSO and GS, improves the classification quality of the ANN.

Using CICIDS2017 dataset, an approach is proposed in Kurniabudi et al. (2020), it combines PSO with an evolutionary algorithm as a feature selection method. The evaluation is performed with the C4.5 tree-based algorithm. The dataset groups the attacks by similarities, relabeling the flows in six new classes, this reduces the chance of wrong classification without loss all attacks detection. This method is able to classify more accurately those grouped attack classes, reducing the false positives and improving the recall and precision, which improves the accuracy metric as well.

Tackling the attack classification from the mcACP problem abstraction, offers some challenges to overcome: data imbalance, a detailed features about classes is required to perform granular separation among the attack classes, the classification algorithms tends a bias behavior to majority classes and is hard to assert a sound multiclass metrics evaluation in all cases. As the multiclass design is quite more complex, the algorithms need to perform a trade-off between model simplicity, performance and approach design complexity. Some approaches combines several techniques, in general those models are hard to implement, explain and often are computational demanding. In the other hand, simpler approaches could achieve good performance in a few particular attacks with a low computational demand, creating high specialized models.

5.5. Advantages, issues and open challenges

In this section we present a summary of techniques advantages, identified issues about the approaches and open challenges regard the analysis of the surveyed works.

One of the most relevant advantage of the wrapper feature selection approaches is its intrinsic flexibility, that allow to combine strategies of different types from the artificial intelligence discipline, which could be implemented with a classical, modified or combined methods. This obey to the goal, rationale and focus of the designers, thus this produce a high flexible and adaptive designs, as the attack detection requires. As the cyberattacks are in constant change, also the research community is, so is a high dynamic environment to develop new strategies oriented to tackle new every day challenges, giving an unlimited research opportunities to the interested communities, from cybersecurity, artificial intelligence, network infrastructure, services, and every discipline that are or want to be involved in this research field. Moreover, with the IoT and software defined networks (SDN) infrastructures, there are a large research field to explore, as those incipient infrastructure provides advantages and security issues that need to be tackle.

The amount of security and monitor data generated by the service infrastructure, is diverse and very large, so the machine learning discipline provides a wide variety of tools to analyze large amount of data, which helps the attack detection behaviors. Moreover, this tools could lead to recognize an unseen anomalous behaviors with a certain precision, thus those tools in combination with other feature search related metrics, provides a framework to build a promising approaches and even to develop new detection strategies, according to the last security environments and applications.

In the other hand, we found that this flexibility advantage, is also a disadvantage, due to the approach design could become complex, hard to explain, interpret and, more important, evaluate. A combination of strategies also is a combination of the advantages and issues of each individual techniques, so the designers must pay attention of this kind of combinations, make a detailed rationale about what is the focus of the research and how much complexity is it willing to pay. So, a full analysis of the problem and the experimental view is necessary to achieve effective approach, also simpler and easy-to-explain as possible.

We notice an important amount of metrics, much of them called with different names depending the discipline, but also much of them evaluate different aspects of the strategies. This situation could lead to a non-comparable results, which can be a disadvantage when the proposed approach need to be sound compared with other existing ones. So, is an important consideration which of those metrics will the researcher use to evaluate an approach and if its fair comparable with the other reference works.

Also the researcher need to carefully select and handle the experimental dataset. Note that, as we seen in our review, there are several datasets to experiment to. So, the experimental dataset and the goal of the research must be aligned and, moreover, is needed to keep in mind if the performed data treatment is comparable with the state of the art. This, in order to make a fully comparable work that contribute with the corpus knowledge of the research area and help other researches in the next approaches comparison and discussion. In the other hand, the data sampling strategy could lead to a biased or incomplete experimental data, which can produce models with an incomplete information. Due to the imbalance nature of the data, the created approaches need to consider the bias of the classification technique, issue that the researchers need to point out in the rationale design.

A general issue found in our revision, is the limited or null experimental reproducibility of the proposed approaches, due to the wide variety of techniques and its construction. Many of them are standard algorithms with a custom modification or a non-standard algorithm implemented to a specific case. The approaches code is often not available or they are not implemented in a public library, reachable to the research community.

Finally, we could identify some open issues, which are research chances. We notice that the experimental dataset need a continuous update. To produce a labeled experimental data is difficult, due to the complex task to design an scenario or environment that reflects the behavior to be modeled, then capture the data, process the captured data into a desired characterization, anonymize it and label the events in it corresponding class and, finally, perform the dataset assessment and publish in the scientific community. The gap between the idea to build a new dataset and it publishing, is high enough to leave outdated the time that this new dataset become public. So, as community, we need to create a methodology that could reduce this gap, in order to have more updated datasets in a shorten time, thus we can propose new approaches closer to current scenarios, providing more accurate solutions to current real applications.

We observe that there are an heterogeneous ways to evaluate the approaches, making difficult to compare them or even to choose a sound evaluation way. In order to tackle that, the community need a methodology that describes a clear steps or a guide to perform an approaches evaluation, that propose guidelines from the dataset treatment, metrics applications, experiment execution, results presentation, among other useful considerations, all of those from a technical

point of view, with the main collaboration of the machine learning and cybersecurity communities. This guidelines and consensus will help to reduce the gap between the new approaches proposal and its application and comparison with the state of the art.

6. Conclusion

The main idea of this work is to analyze and point out the characteristics, advantages, issues and open challenges of the most recent wrapper feature selection approaches, in order to give a start point to the interested researchers in this area. Note that we perform a comparative of the surveyed work, considering a fair comparison regard the dataset, reported metrics and pointing out the most relevant design characteristics of the approach, in order to provide brief but relevant information about all the approaches, giving a breadth view of the state of the art.

We found key characteristics to consider in the wrapper approaches design, oriented to tackle the still open and up-to-date problem of intrusion detection. It is a constantly changing and very complex problem that comes from the cybersecurity discipline to the research community of artificial intelligence and it always require attention, as the technology advance and the continuous every day operations convergence on IT infrastructures. Observed facts as the complexity of the problem, the vast amount of available techniques, the possible sources of testing available data, the growing IoT technologies and new IT, all make this problem very active and interesting to the cybersecurity and artificial intelligence research communities.

We notice several approaches that combine bio-inspired and machine learning techniques, taking the techniques advantage of the combinatorial handle and the power of classification, respectively. Many of them aim to deal with the feature selection as a maximization (using DR, Precision, Accuracy measures) or minimization (using misclassification, FPR, FAR measures) problem, tackled from bACP of mcACP perspective. We note that there are different ways to measure the quality of the approaches, and it is difficult to compare them in a precise way. Some of them use the well-documented and classical indicators (such as DR, FPR, Acc, PR); others create a combination of them, designed to consider as many relevant aspects of the problem as possible. Also, there are approaches that create their own custom indicators which makes difficult to perform a fair comparison.

We propose a taxonomy based on design details of the wrapper feature selection approaches and group the evaluation metrics according of two observed abstraction of the problem, such bACP and mcACP, moreover we remark the differences about such modes. All this with the idea of help in the design of components that help to build approaches with a breath view and considerations, thus a better understand of the construction and evaluation of the approaches.

There is a significant and growing amount of datasets, designed for different aspects of the attack detection problem. Some of those datasets are criticable (built from an simulated data, biased to a key aspect, private) or outdated, but are very well documented and widely used, presenting classical attacks (such KDD99 and NSL-KDD), while others are oriented to more recent network activity and current attacks (such as CICIDS2017 and ISCX2012). These datasets are created from real or synthetic data, labeled and designed to be used by the research community, considering aspects such as attack class, attack scenarios (a term from recent multi-steps attack), flow types, descriptive features, time windows, etc, which could be used at the discretion of the researchers. Nevertheless, there is an important gap regard the scientific available datasets and the real world scenario data, this is due to the research community need to perform several steps to produce a new dataset from a real or updated scenario.

Due to highly changing attacks morphology, an interesting area to explore in future work is the so-called zero day attacks or unknown attacks, those which do not have any information yet and, for this reason, are very difficult (or close to impossible) to detect or even unnoticeable if they occur. In this context fall the new so-called multi-step attacks or attack scenario, which are more complex and difficult to detect, because they are composed of actions that are not necessarily harmful or suspicious, which could be performed in different sequences.

There are plenty of work to do, as we describe, these approaches need a consensus about how to propose a unified evaluation methodology that allow to the researchers show its results in a sound comparative way, makes easier to contribute with new approaches that aim to solve still open issues. Moreover, the community need to close the gap between the new behavior apparition and the scientific response, this in order to advance efficiently in new develops fast as they can, to mitigate the risks of unknown attacks.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was supported by Fondecyt, Chile under the grant 1200126.

References

Abbasi, M. S., Al-Sahaf, H., & Welch, I. (2020). Particle swarm optimization: A wrapper-based feature selection method for ransomware detection and classification. In *Applications of Evolutionary Computation* (pp. 181–196). Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-43722-0_12.

Abdullah, M., Alshannaq, A., Balamash, A., & Almabdy, S. (2018). Enhanced intrusion detection system using feature selection method and ensemble learning algorithms. *International Journal of Computer Science and Information Security (IJCSIS)*, *16*(2).

Ahmim, A., Derdour, M., & Ferrag, M. A. (2018). An intrusion detection system based on combining probability predictions of a tree of classifiers. *International Journal of Communication Systems*, *31*(9), 1–17. http://dx.doi.org/10.1002/dac.3547.

Alazzam, H., Sharieh, A., & Sabri, K. E. (2020). A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer. *Expert Systems with Applications*, *148*, http://dx.doi.org/10.1016/j.eswa.2020.113249.

Aljawarneh, S., Aldwairi, M., & Yassein, M. B. (2016). Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computer Science*, *25*, 152–160. http://dx.doi.org/10.1016/j.jocs.2017.03.006, URL: https://www.sciencedirect.com/science/article/pii/S1877750316305099?via%3Dihub.

Almasoudy, F. H., Al-yaseen, W. L., & Idrees, A. K. (2020). Differential evolution wrapper feature selection for intrusion detection system. *Procedia Computer Science*, *167*(2019), 1230–1239. http://dx.doi.org/10.1016/j.procs.2020.03.438.

Almomani, O. (2020). A feature selection model for network intrusion detection system based on PSO, GWO, FFA and GA algorithms. *Symmetry*, *12*(6), 1046.

Anwer, H. M., Farouk, M., & Abdel-Hamid, A. (2018). A framework for efficient network anomaly intrusion detection with features selection. In *2018 9th International Conference on Information and Communication Systems, ICICS 2018, Vol. 2018-Janua* (pp. 157–162). IEEE, http://dx.doi.org/10.1109/IACS.2018.8355459.

Arnaldo, I., & Lam, M. (2019). EX 2 : a framework for interactive anomaly detection. In C. Trattner, D. Parra, & N. Richie (Eds.), *Joint Proceedings of the ACM IUI 2019 Workshops* (p. 5). Los Angeles, USA: CEUR-WS.org.

Ashfaq, R. A. R., Wang, X. Z., Huang, J. Z., Abbas, H., & He, Y. L. (2017). Fuzziness based semi-supervised learning approach for intrusion detection system. *Information Sciences*, *378*, 484–497. http://dx.doi.org/10.1016/j.ins.2016.04.019.

Balasaraswathi, V. R., Sugumaran, M., & Hamid, Y. (2017). Feature selection techniques for intrusion detection using non-bio-inspired and bio-inspired optimization algorithms. *Journal of Communications and Information Networks*, *2*(4), 107–119. http://dx.doi.org/10.1007/s41650-017-0033-7.

Bao, Z., Muñoz González, L., & Lupu, E. C. (2018). Mitigation of adversarial attacks through embedded feature selection. (p. 12). arXiv preprint arXiv:1808.05705.

Botes, F. H., Leenen, L., & De La Harpe, R. (2017). Ant colony induced decision trees for intrusion detection. In *European Conference on Information Warfare and Security, ECCWS* (pp. 53–62). Elsevier Scopus.

Bouzoubaa, K., Taher, Y., & Nsiri, B. (2021). Predicting DOS-DDOS attacks: Review and evaluation study of feature selection methods based on wrapper process. *International Journal of Advanced Computer Science and Applications*, *12*(5), 132–145. http://dx.doi.org/10.14569/IJACSA.2021.0120517.

Bridges, R. A., Glass-Vanderlan, T. R., Iannacone, M. D., Vincent, M. S., & Chen, Q. (2019). A survey of intrusion detection systems leveraging host data. *ACM Computing Surveys*, *52*(6), 1–40. http://dx.doi.org/10.1145/3344382, arXiv:1805.06070.

Cai, J., Luo, J., Wang, S., & Yang, S. (2018). Feature selection in machine learning: A new perspective. *Neurocomputing*, *300*, 70–79. http://dx.doi.org/10.1016/j.neucom.2017.11.077.

Cisco Systems (2018). *Cisco 2018 Annual Cybersecurity Report*: *Technical Report Cisco Systems*, (p. 65). http://dx.doi.org/10.1002/ejoc.201200111, URL: https://www.cisco.com/c/dam/m/hu_hu/campaigns/security-hub/pdf/acr-2018.pdf.

da Costa, K. A., Papa, J. P., Lisboa, C. O., Munoz, R., & de Albuquerque, V. H. C. (2019). Internet of things: A survey on machine learning-based intrusion detection approaches. *Computer Networks*, *151*, 147–157. http://dx.doi.org/10.1016/j.comnet.2019.01.023, URL: https://doi.org/10.1016/j.comnet.2019.01.023.

Creech, G., & Hu, J. (2013). Generation of a new IDS test dataset: Time to retire the kdd collection. In *IEEE Wireless Communications and Networking Conference, WCNC* (pp. 4487–4492). IEEE, http://dx.doi.org/10.1109/WCNC.2013.6555301.

Davahli, A., Shamsi, M., & Abaei, G. (2020). A lightweight anomaly detection model using SVM for WSNs in IoT through a hybrid feature selection algorithm based on GA and. *Journal of Computing and Security ALightweight*, *7*(1), 63–79.

Divyasree, T. H., & Sherly, K. K. (2018). A network intrusion detection system based on ensemble CVM using efficient feature selection approach. *Procedia Computer Science*, *143*, 442–449. http://dx.doi.org/10.1016/j.procs.2018.10.416.

Elhag, S., Fernández, A., Bawakid, A., Alshomrani, S., & Herrera, F. (2015). On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems. *Expert Systems with Applications*, *42*(1), 193–202. http://dx.doi.org/10.1016/j.eswa.2014.08.002.

Gan, J., Wen, G., Yu, H., Zheng, W., & Lei, C. (2020). Supervised feature selection by self-paced learning regression. *Pattern Recognition Letters*, *132*, 30–37. http://dx.doi.org/10.1016/j.patrec.2018.08.029.

Gauthama Raman, M. R., Somu, N., Kirthivasan, K., Liscano, R., & Shankar Sriram, V. S. (2017). An efficient intrusion detection system based on hypergraph - genetic algorithm for parameter optimization and feature selection in support vector machine. *Knowledge-Based Systems*, *134*, 1–12. http://dx.doi.org/10.1016/j.knosys.2017.07.005, URL: https://www.sciencedirect.com/science/article/pii/S0950705117303209.

Gharaee, H., Fekri, M., & Hosseinvand, H. (2018). Intrusion detection system using SVM as classifier and GA for optimizing feature vectors. *International Journal of Information & Communication Technology Research (IJICTR)*, *10*(1), 26–35.

Golrang, A., Golrang, A. M., Yayilgan, S. Y., & Elezaj, O. (2020). A novel hybrid ids based on modified NSGAII-ANN and random forest. *Electronics (Switzerland)*, *9*(4), 1–19. http://dx.doi.org/10.3390/electronics9040577.

Grandini, M., Bagli, E., & Visani, G. (2020). Metrics for multi-class classification: an overview. (pp. 1–17). URL: http://arxiv.org/abs/2008.05756.

Gringoli, F., Salgarelli, L., Dusi, M., Cascarano, N., Risso, F., et al. (2009). Gt: picking up the truth from the ground for internet traffic. *ACM SIGCOMM Computer Communication Review*, *39*(5), 12–18.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*(2), 1157–1182. http://dx.doi.org/10.1016/j.aca.2011.07.027.

Hajisalem, V., & Babaie, S. (2018). A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection. *Computer Networks*, *136*, 37–50. http://dx.doi.org/10.1016/j.comnet.2018.02.028.

Hamed, T., Dara, R., & Kremer, S. C. (2018). Network intrusion detection system based on recursive feature addition and bigram technique. *Computers and Security*, *73*, 137–155. http://dx.doi.org/10.1016/j.cose.2017.10.011.

Hindy, H., Brosset, D., Bayne, E., Seeam, A., Tachtatzis, C., Atkinson, R., et al. (2018). A taxonomy and survey of intrusion detection system design techniques, network threats and datasets. *CoRR*, *1*(1), 1–35, arXiv:1806.03517.

Hosseini, S. (2020). A new machine learning method consisting of GA-LR and ANN for attack detection. *Wireless Networks*, *26*(6), 4149–4162. http://dx.doi.org/10.1007/s11276-020-02321-3.

Hosseini, S., & Zade, B. M. H. (2020). New hybrid method for attack detection using combination of evolutionary algorithms, SVM, and ANN. *Computer Networks*, *173*(February 2019), Article 107168. http://dx.doi.org/10.1016/j.comnet.2020.107168, URL: https://doi.org/10.1016/j.comnet.2020.107168.

Hosseini Bamakan, S. M., Wang, H., Yingjie, T., & Shi, Y. (2016). An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization. *Neurocomputing*, *199*, 90–102. http://dx.doi.org/10.1016/j.neucom.2016.03.031, URL: https://www.sciencedirect.com/science/article/pii/S0925231216300510?via%3Dihub.

International Organization for Standardization (2018). International standard iso / iec information technology — security techniques — information security management systems — requirements. *Information Technology — Security Techniques — Information Security Management Systems — Requirements*, *2018–07*(ISO/IEC 27001:2013), 38.

Jiang, B., Li, C., De Rijke, M., Yao, X., & Chen, H. (2019). Probabilistic feature selection and classification vector machine. *ACM Transactions on Knowledge Discovery from Data*, *13*(2), http://dx.doi.org/10.1145/3309541, arXiv:1609.05486.

Jiang, S., & Xu, X. (2019). Impact of feature selection methods on data classification for IDS. In *Proceedings - 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2019* (pp. 174–180). IEEE, http://dx.doi.org/10.1109/CyberC.2019.00039.

Jiang, J., Yu, Q., Yu, M., Li, G., Chen, J., Liu, K., et al. (2018). ALDD: A hybrid traffic-user behavior detection method for application layer DDoS. In *Proceedings - 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications and 12th IEEE International Conference on Big Data Science and Engineering, Trustcom/BigDataSE 2018* (pp. 1565–1569). IEEE, http://dx.doi.org/10.1109/TrustCom/BigDataSE.2018.00225.

Kalaivani, S., & Gopinath, G. (2020). Modified bee colony with bacterial foragin optimization based hybrid feature selection technique for intrusion detection system classifier model. *ICTACT Journal on Soft Computing*, *10*(4), 2146–2152. http://dx.doi.org/10.21917/ijsc.2020.0305.

Kamarudin, M. H., Maple, C., & Watson, T. (2019). Hybrid feature selection algorithm for intrusion detection system. *International Journal of High Performance Computing and Networking*, *13*, 232–240. http://dx.doi.org/10.3844/jcssp.2014.1015.1025.

Kamarudin, M. H., Maple, C., Watson, T., & Safa, N. S. (2017). A LogitBoost-based algorithm for detecting known and unknown web attacks. *IEEE Access*, *5*, 26190–26200. http://dx.doi.org/10.1109/ACCESS.2017.2766844, URL: http://ieeexplore.ieee.org/document/8094857/.

Kasongo, S. M., & Sun, Y. (2020). A deep learning method with wrapper based feature extraction for wireless intrusion detection system. *Computers and Security*, *92*, http://dx.doi.org/10.1016/j.cose.2020.101752.

Kayacik, H. G., Zincir-Heywood, A. N., & Heywood, M. I. (2005). Selecting features for intrusion detection: A feature relevance analysis on KDD 99 intrusion detection datasets. In *Proceedings of the Third Annual Conference on Privacy, Security and Trust*.

Khammassi, C., & Krichen, S. (2020). A NSGA2-LR wrapper approach for feature selection in network intrusion detection. *Computer Networks*, *172*(November 2019), Article 107183. http://dx.doi.org/10.1016/j.comnet.2020.107183.

Khorram, T. (2020). Feature selection in network intrusion detection using metaheuristic algorithms. *International Journal of Advance Research, Ideas and Innovations in Technology*, *4*(4), 704–710.

Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*, *2*(1), 22. http://dx.doi.org/10.1186/s42400-019-0038-7.

Kohavi, R., & John, G. (1997). Wrappers for feature subset selection. In *7920 LNCS*, *Artificial Intelligence 97* (pp. 273–324). http://dx.doi.org/10.1007/978-3-642-39038-8-27.

Kolias, C., Kambourakis, G., Stavrou, A., & Gritzalis, S. (2016). Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset. *IEEE Communications Surveys & Tutorials*, *18*(1), 184–208. http://dx.doi.org/10.1109/COMST.2015.2402161.

Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (2019). Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems*, *100*, 779–796, arXiv:1811.00701.

Kozik, R., & Chora, M. (2019). Balanced efficient lifelong learning (B-ELLA) for cyber attack detection. *Journal of Universal Computer Science*, *25*(1), 2–15.

Kumar, B. N., Raju, M. S. V. S., & Vardhan, B. V. (2018). A comparative survey on the influence of machine learning techniques on intrusion detection system (IDS). *IOSR Journal of Engineering (IOSRJEN)*, *08*(8), 25–42.

Kurniabudi, Harris, A., Mintaria, A. E., Darmawijoyo, Stiawan, D., bin Idris, M. Y., et al. (2020). Improving the anomaly detection by combining PSO search methods and J48 algorithm. In *EECSI 2020* (October), (pp. 119–126).

Latha, S. (2018). HPFSM - a high pertinent feature selection mechanism for intrusion detection system. *International Journal of Pure and Applied Mathematics*, *118*(9), 77–83.

Li, H., Guo, W., Wu, G., & Li, Y. (2018). A RF-PSO based hybrid feature selection model in intrusion detection system. In *Proceedings - 2018 IEEE 3rd International Conference on Data Science in Cyberspace, DSC 2018* (pp. 795–802). IEEE, http://dx.doi.org/10.1109/DSC.2018.00128.

Lippmann, R. P., Fried, D. J., Graf, I., Haines, J. W., Kendall, K. R., McClung, D., et al. (2000). Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation. *2*, In *Proceedings - DARPA Information Survivability Conference and Exposition, DISCEX 2000* (pp. 12–26). http://dx.doi.org/10.1109/DISCEX.2000.821506.

Liu, H., & Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. *Applied Sciences (Switzerland)*, *9*(20), http://dx.doi.org/10.3390/app9204396.

Liu, Z., & Shi, Y. (2022). A hybrid IDS using GA-based feature selection method and the random forest. *International Journal of Machine Learning and Computing*, *12*(2), 1–14. http://dx.doi.org/10.18178/ijmlc.2022.12.2.1077.

Liu, Z., Wang, R., Tao, M., & Cai, X. (2015). A class-oriented feature selection approach for multi-class imbalanced network traffic datasets based on local and global metrics fusion. *Neurocomputing*, *168*, 365–381. http://dx.doi.org/10.1016/j.neucom.2015.05.089, URL: http://dx.doi.org/10.1016/j.neucom.2015.05.089.

Maldonado, J., & Riff, M.-C. (2019). Evaluating different metric configurations of an evolutionary wrapper for attack detection. In *31st International Conference on Tools with Artificial Intelligence ICTAI* (pp. 1–5).

Maldonado, J., & Riff, M.-C. (2020). Improving an evolutionary wrapper for attack detection by including feature importance information. In *GECCO '20*, *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion* (pp. 139–140). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3377929.3390021, URL: https://doi-org.usm.idm.oclc.org/10.1145/3377929.3390021.

Maldonado, J., Riff, M.-C., & Montero, E. (2019). Improving attack detection of C4.5 using an evolutionary algorithm. In *2019 IEEE Congress on Evolutionary Computation (CEC)* (pp. 2229–2235). IEEE, http://dx.doi.org/10.1109/CEC.2019.8790199, URL: https://ieeexplore.ieee.org/document/8790199/.

Mazini, M., Shirazi, B., & Mahdavi, I. (2018). 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*, *31*(4), 541–553. http://dx.doi.org/10.1016/j.jksuci.2018.03.011, URL: https://doi.org/10.1016/j.jksuci.2018.03.011.

Milenkoski, A., Vieira, M., Kounev, S., Avritzer, A., & Payne, B. D. (2015). Evaluating computer intrusion detection systems: A survey of common practices. *ACM Computing Surveys*, *48*(1), 41. http://dx.doi.org/10.1145/2808691.

Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E. S. (2019). A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Communications Surveys & Tutorials*, *21*(1), 686–728. http://dx.doi.org/10.1109/COMST.2018.2847722, URL: https://ieeexplore.ieee.org/document/8386762/.

Mohammadi, S., Desai, V., & Karimipour, H. (2018). Multivariate mutual information-based feature selection for cyber intrusion detection. In *2018 IEEE Electrical Power and Energy Conference, EPEC 2018* (pp. 1–6). IEEE, http://dx.doi.org/10.1109/EPEC.2018.8598326.

Mohammadi, S., Mirvaziri, H., Ghazizadeh-Ahsaee, M., & Karimipour, H. (2019). Cyber intrusion detection by combined feature selection algorithm. *Journal of Information Security and Applications*, *44*, 80–88. http://dx.doi.org/10.1016/j.jisa.2018.11.007, URL: https://doi.org/10.1016/j.jisa.2018.11.007.

Mohd Yusof, M. H., Mokhtar, M. R., Zain, A. M., & Maple, C. (2018). Embedded feature selection method for a network-level behavioural analysis detection model. *International Journal of Advanced Computer Science and Applications*, *9*(12), 509–517. http://dx.doi.org/10.14569/IJACSA.2018.091271.

Moore, A. W., & Zuev, D. (2005). Internet class moore and zuev. In *SIGMETRICS* (pp. 50–60). ACM.

Moustafa, N., & Slay, J. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference (MilCIS)* (pp. 1–6).

Nancy, P., Muthurajkumar, S., Ganapathy, S., Santhosh Kumar, S. V., Selvi, M., & Arputharaj, K. (2020). Intrusion detection using dynamic feature selection and fuzzy temporal decision tree classification for wireless sensor networks. *IET Communications*, *14*(5), 888–895. http://dx.doi.org/10.1049/iet-com.2019.0172.

Navarro, J., Deruyver, A., & Parrend, P. (2018). A systematic survey on multi-step attack detection. *Computers & Security*, *76*, 214–249. http://dx.doi.org/10.1016/j.cose.2018.03.001, URL: http://linkinghub.elsevier.com/retrieve/pii/S0167404818302141.

Navarro, J., Legrand, V., Deruyver, A., & Parrend, P. (2018). OMMA: open architecture for operator-guided monitoring of multi-step attacks. *Eurasip Journal on Information Security*, *2018*(1), 6. http://dx.doi.org/10.1186/s13635-018-0075-x, URL: https://jis-eurasipjournals.springeropen.com/articles/10.1186/s13635-018-0075-x.

Navarro-Lara, J., Deruyver, A., & Parrendz, P. (2017). Morwilog: An ACO-based system for outlining multi-step attacks. In *2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016* (pp. 1–8). IEEE, http://dx.doi.org/10.1109/SSCI.2016.7849902, URL: http://ieeexplore.ieee.org/document/7849902/.

Nazir, A., & Khan, R. A. (2019). Combinatorial optimization based feature selection method: A study on network intrusion detection. (pp. 1–27). ArXiv, URL http://arxiv.org/abs/1906.04494.

Nazir, A., & Khan, R. A. (2021). A novel combinatorial optimization based feature selection method for network intrusion detection. *Computers and Security*, *102*, Article 102164. http://dx.doi.org/10.1016/j.cose.2020.102164.

Nisioti, A., Mylonas, A., Yoo, P. D., & Katos, V. (2018). From intrusion detection to attacker attribution: A comprehensive survey of unsupervised methods. *IEEE Communications Surveys & Tutorials*, *20*(4), 3369–3388. http://dx.doi.org/10.1109/COMST.2018.2854724.

Pendleton, M., Garcia-Lebron, R., Cho, J. H., & Xu, S. (2016). A survey on systems security metrics. *ACM Computing Surveys*, *49*(4), 35. http://dx.doi.org/10.1145/3005714.

Polat, H., & Polat, O. (2020). Detecting ddos attacks in software-defined networks through feature selection methods and machine learning models. *Sustainability*, *3*(12), 16.

Resende, P. A. A., & Drummond, A. C. (2018a). A survey of random forest based methods for intrusion detection systems. *ACM Computing Surveys*, *51*(3), http://dx.doi.org/10.1145/3178582.

Resende, P. A. A., & Drummond, A. C. (2018b). Adaptive anomaly-based intrusion detection system using genetic algorithm and profiling. *Security and Privacy*, *1*(4), Article e36. http://dx.doi.org/10.1002/spy2.36, URL: http://doi.wiley.com/10.1002/spy2.36.

Ring, M., Wunderlich, S., Scheuring, D., Landes, D., & Hotho, A. (2019). A survey of network-based intrusion detection data sets. *Computers and Security*, *86*, 147–167. http://dx.doi.org/10.1016/j.cose.2019.06.005, arXiv:1903.02460.

Roman, R., Lopez, J., & Mambo, M. (2018). Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges. *Future Generation Computer Systems*, *78*, 680–698. http://dx.doi.org/10.1016/j.future.2016.11.009.

Salo, F., Nassif, A. B., & Essex, A. (2019). Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection. *Computer Networks*, *148*, 164–175. http://dx.doi.org/10.1016/j.comnet.2018.11.010.

Saloky, T., & Šeminský, J. (2017). Artificial intelligence and machine learning applied to cybersecurity. In *IEEE Confluence*: *Technical Report*, (pp. 1–18). Philadelphia, PA, USA: IEEE, URL: http://uni-obuda.hu/conferences/SAMI2005/SALOKY.pdf.

Samadi Bonab, M., Ghaffari, A., Soleimanian Gharehchopogh, F., & Alemi, P. (2020). A wrapper-based feature selection for improving performance of intrusion detection systems. *International Journal of Communication Systems*, *33*(12), 1–25. http://dx.doi.org/10.1002/dac.4434.

Sarıkaya, A., & Kılıç, B. G. (2020). A class-specific intrusion detection model: Hierarchical multi-class IDS model. *SN Computer Science*, *1*(4), 1–11. http://dx.doi.org/10.1007/s42979-020-00213-z.

Sarvari, S., Mohd Sani, N. F., Mohd Hanapi, Z., & Abdullah, M. T. (2020). An efficient anomaly intrusion detection method with feature selection and evolutionary neural network. *IEEE Access*, *8*, 70651–70663. http://dx.doi.org/10.1109/ACCESS.2020.2986217.

Selvakumar, B., & Muneeswaran, K. (2019). Firefly algorithm based feature selection for network intrusion detection. *Computers and Security*, *81*, 148–155. http://dx.doi.org/10.1016/j.cose.2018.11.005.

Sen, S. (2015). *A Survey of Intrusion Detection Systems using Evolutionary Computation* (pp. 73–94). Elsevier Inc., http://dx.doi.org/10.1016/B978-0-12-801538-4.00004-5.

Sgandurra, D., Muñoz González, L., Mohsen, R., & Lupu, E. C. (2016). Automated dynamic analysis of ransomware: Benefits, limitations and use for detection. CoRR, URL: http://arxiv.org/abs/1609.03020.

Shafiq, M., Tian, Z., Bashir, A. K., Du, X., & Guizani, M. (2020). IoT malicious traffic identification using wrapper-based feature selection mechanisms. *Computers and Security*, *94*, http://dx.doi.org/10.1016/j.cose.2020.101863.

Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy* (Cic), (pp. 108–116). http://dx.doi.org/10.5220/0006639801080116, URL: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006639801080116.

Shiravi, A., Shiravi, H., Tavallaee, M., & Ghorbani, A. A. (2012). Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Computers and Security*, *31*(3), 357–374. http://dx.doi.org/10.1016/j.cose.2011.12.012.

Soheily-Khah, S., Marteau, P. F., & Bechet, N. (2018). Intrusion detection in network systems through hybrid supervised and unsupervised machine learning process: A case study on the iscx dataset. In *Proceedings - 2018 1st International Conference on Data Intelligence and Security, ICDIS 2018* (pp. 219–226). http://dx.doi.org/10.1109/ICDIS.2018.00043.

Solani, S., & Jadav, N. K. (2021). *A Novel Approach to Reduce False-Negative Alarm Rate in Network-Based Intrusion Detection System using Linear Discriminant Analysis, Vol. 145* (pp. 911–921). Springer Singapore, http://dx.doi.org/10.1007/978-981-15-7345-3_77.

Solorio-Fernández, S., Carrasco-Ochoa, J. A., & Martínez-Trinidad, J. F. (2020). A review of unsupervised feature selection methods. *Artificial Intelligence Review*, *53*(2), 907–948. http://dx.doi.org/10.1007/s10462-019-09682-y.

Stallings, W., Brown, L., Bauer, M. D., & Bhattacharjee, A. K. (2012). *Computer security: principles and practice*. Pearson Education.

Stellios, I., Kotzanikolaou, P., Psarakis, M., Alcaraz, C., & Lopez, J. (2018). A survey of iot-enabled cyberattacks: Assessing attack paths to critical infrastructures and services. *IEEE Communications Surveys & Tutorials*, *20*(4), 3453–3495. http://dx.doi.org/10.1109/COMST.2018.2855563.

Su, Y.-J., Huang, P.-Y., Hu, W.-C., Lin, H.-Y., Kao, C.-Y., Hsieh, S.-H., et al. (2019). Using feature selection to improve performance of three-tier intrusion detection system. *Advances in Intelligent Systems and Computing*, *927*(October), 776–785. http://dx.doi.org/10.1007/978-3-030-15035-8, URL: http://dx.doi.org/10.1007/978-3-030-15035-8{_}93%0Ahttp://link.springer.com/10.1007/978-3-030-15035-8.

Syarif, I., Afandi, R. F., & Astika Saputra, F. (2020). Feature selection algorithm for intrusion detection using cuckoo search algorithm. In *2020 International Electronics Symposium (IES)* (pp. 430–435). IEEE, http://dx.doi.org/10.1109/IES50839.2020.9231840, URL: https://ieeexplore.ieee.org/document/9231840/.

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In *IEEE Symposium on Computational Intelligence for Security and Defense Applications, CISDA 2009* (Cisda), (pp. 1–6). IEEE, http://dx.doi.org/10.1109/CISDA.2009.5356528.

Tharwat, A. (2018). Classification assessment methods. *Applied Computing and Informatics*, http://dx.doi.org/10.1016/j.aci.2018.08.003.

Tounsi, W., & Rais, H. (2018). A survey on technical threat intelligence in the age of sophisticated cyber attacks. *Computers and Security*, *72*, 212–233. http://dx.doi.org/10.1016/j.cose.2017.09.001.

Umar, M. A., Zhanfang, C., & Liu, Y. (2020). Network intrusion detection using wrapper-based decision tree for feature selection. In *International Conference on Internet Computing for Science and Engineering* (pp. 5–13). http://dx.doi.org/10.1145/3424311.3424330, arXiv:2008.07405.

Venkatesh, B., & Anuradha, J. (2019). A review of feature selection and its methods. *Cybernetics and Information Technologies*, *19*(1), 3–26. http://dx.doi.org/10.2478/CAIT-2019-0001.

Vijayanand, R., & Devaraj, D. (2020). A novel feature selection method using whale optimization algorithm and genetic operators for intrusion detection system in wireless mesh network. *IEEE Access*, *8*, 56847–56854. http://dx.doi.org/10.1109/access.2020.2978035.

Vijayanand, R., Devaraj, D., & Kannapiran, B. (2018). Intrusion detection system for wireless mesh network using multiple support vector machine classifiers with genetic-algorithm-based feature selection. *Computers & Security*, *77*, 304–314. http://dx.doi.org/10.1016/j.cose.2018.04.010, URL: https://www.sciencedirect.com/science/article/pii/S0167404818303766?via%3Dihub, https://linkinghub.elsevier.com/retrieve/pii/S0167404818303766.

Wahba, Y., ElSalamouny, E., & ElTaweel, G. (2015). Improving the performance of multi-class intrusion detection systems using feature reduction. *International Journal of Computer Science*, *12*(3), 255–262, URL: http://arxiv.org/abs/1507.06692.

Wang, M., Lu, Y., & Qin, J. (2020). A dynamic MLP-based ddos attack detection method using feature selection and feedback. *Computers and Security*, *88*, http://dx.doi.org/10.1016/j.cose.2019.101645.

Wei, W., Chen, S., Lin, Q., Ji, J., & Chen, J. (2020). A multi-objective immune algorithm for intrusion feature selection. *Applied Soft Computing*, *95*, Article 106522. http://dx.doi.org/10.1016/j.asoc.2020.106522, URL: https://doi.org/10.1016/j.asoc.2020.106522.

Xue, Y., Jia, W., Zhao, X., & Pang, W. (2018). An evolutionary computation based feature selection method for intrusion detection. *Security and Communication Networks*, *2018*, 1–10. http://dx.doi.org/10.1155/2018/2492956.

Xue, B., Zhang, M., Browne, W. N., & Yao, X. (2016). A survey on evolutionary computation approaches to feature selection. *IEEE Transactions on Evolutionary Computation*, *20*(4), 606–626. http://dx.doi.org/10.1109/TEVC.2015.2504420, arXiv:1612.08669.

Yılmaz Gündüz, S., & Çeter, M. N. (2018). Feature selection and comparison of classification algorithms for intrusion detection. *Anadolu University Journal of Science and Technology - Applied Sciences and Engineering*, *19*(1), 206–218. http://dx.doi.org/10.18038/aubtda.356705.

Zhang, Q., Qu, Y., & Deng, A. (2018). Network intrusion detection using kernel-based fuzzy-rough feature selection. *2018-July*, In *IEEE International Conference on Fuzzy Systems* (pp. 1–6). IEEE, http://dx.doi.org/10.1109/FUZZ-IEEE.2018.8491578.

Zhou, Y., Cheng, G., Jiang, S., & Dai, M. (2020). Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Computer Networks*, Article 107247. http://dx.doi.org/10.1016/j.comnet.2020.107247, arXiv:1904.01352.

Zhu, L., He, S., Wang, L., Zeng, W., & Yang, J. (2019). Feature selection using an improved gravitational search algorithm. *IEEE Access*, *7*, 114440–114448. http://dx.doi.org/10.1109/ACCESS.2019.2935833.

Javier Maldonado

María Cristina Riff

Bertrand Neveu