Data Analysis of Wireless Networks Using Computational Intelligence

Daniel R. Canêdo[1,2] and Alexandre R. S. Romariz[1]
[1] Universidade de Brasília-UnB/Departamento de Engenharia Elétrica, Brasília, Brazil
[2] Instituto Federal de Goiás - IFG, Luziânia, Brazil
Email: daniel.canedo@ifg.edu.br; romariz@ene.unb.br

Abstract**—**In the last decade a great technological advance in mobile technologies infrastructure was seen. The increase in the use of wireless local networks and the use of services from satellites is also noticed. The high utilization rate of mobile devices for various purposes makes clear the need to monitor wireless networks to ensure the integrity and confidentiality of the information transmitted. Therefore, it is necessary to quickly and efficiently identify the normal and abnormal traffic of these networks, so that the administrators can take action. This work aims, from a database of wireless networks, to classify this data in some classes of pre-established anomalies according to some defined criteria of the MAC layer, using supervised and unsupervised intelligent algorithms Multilayer Perceptron (MLP), K-Means and Self-Organizing Maps (SOM). For the analysis of the mentioned algorithms, the WEKA Data Mining software (Waikato Environment for Knowledge Analysis) is used. The algorithms have high success rate in the classification of the data, being indicated in the use of Intrusion Detection Systems for Wireless Networks.

Index Terms**—**Wireless networks, multilayer perceptron, K-means, self-organized map, weka

I. Introduction

In the last decade a great technological advance was seen, especially regarding mobile technologies and its infrastructure. The increase in the use of wireless local networks and also the use of services from satellites, both in organizational and residential environments, is identified. This allows information to be created, transmitted and accessed faster and anywhere at any time by simply having access to the mobile network infrastructure. According to Anatel (Telecommunication National Agency), in January/2016 Brazil registered 257.248 million active lines in mobile telephony, with pre-paid accesses corresponding to 71.45% (183.80 million) of total accesses, while postpaid accesses correspond to 28.55% (73.45 million).

The consequence of this scenario is perceived when the use of computational devices used by both individuals and companies is verified. This scenario can be verified through the research conducted by IDC Brasil, which states that in the last quarter of 2014 Brazil had 1,637 million computers, of which 600 thousand are desktops and 1,037 million are notebooks. An unpublished survey by the Brazilian Institute of Geography and Statistics (IBGE) reveals that 57.3% of homes access the internet through cell phones and tablets in the year 2013.

The Wireless Networks environment, as well as the environment of Ad Hoc Wireless Networks or Wireless Sensor Networks, has in its characteristic a dynamicity in relation to the composition of the network members, that is, for these types of networks users often enter and leave the network. This feature makes real-time management of these environments necessary. This scenario becomes, however, quite vulnerable to attempts to approach the anomalies present in the system as a whole. Anomalies such as *EAPOL Start, Beacon Flood, Deauthentication, RTS Flood* [1].

However, the techniques and tools adopted by network managers in the framework of structured computer networks do not always meet these needs in a timely manner. In this sense, the use of computational intelligence techniques becomes a great option to minimize these difficulties aiming to increasingly identify real-time anomalies.

The high rate of use of mobile devices for various purposes makes clear the need to monitor this infrastructure, since it presents the large-scale transmission of information, which at certain moments may be confidential. The set of this mobile system, determined by both the software and the hardware used, is relatively fragile regarding security, mainly due to the characteristic of its transmission mean, but also due to dynamic access it. So, there is a need to try to quickly and effectively identify the normal and abnormal traffic of these wireless networks so that administrators can take action. This work aims, from a database of wireless networks [1], to classify this data according to some defined criteria of the MAC layer.

The structure of this article is organized into sections. In section two will be presented some works that have the characteristic of identification of wireless networks traffic using algorithms of learning. In section 3, the theoretical basis for Wireless Networks is presented, while section 4 deals with Computational Intelligence Techniques: Neural Networks and K-Means algorithm. Section 5 will present the methodology of experimentation and results. In Section 6 we present the case studies used to analyze the results. In section 7 will be performed the quantitative and qualitative analysis of the results. Section 8 presents the conclusion of the work and future work.

Manuscript received March 22, 2018; revised October 16, 2018.
oi:10.12720/jcm.13.11.618-626

II. Related Works

It is possible to find in the literature some works of Wireless Networks traffic classification, which can be applied in Intrusion Detection Systems. These proposals make use of supervised and unsupervised learning methods. The proposal [2] provides a general approach to the various classification methods, using high-dimensional data and a variable selection technique aiming to reduce computational time and improving the learning rate.

Govindarajan presents a proposal [3] of two classification methods involving multilayer perceptron and Basis function Networks. This work proposes a hybrid architecture involving both classifiers for intrusion detection systems. Ed Wilson presents a proposal [4] of Hybrid Intrusion Detection System, in which signal processing is performed using the Wavelet transform and then the classification of the anomalies using Artificial Neural Networks.

Ed Wilson [1] proposes the elaboration of a real database of Wireless Network traffic, which will be used in the evaluation of Intrusion Detection Systems (IDS). This data, in turn, undergoes a pre-processing to later be classified by techniques of standards recognition, such as Artificial Neural Networks.

III. Wireless Networks

The IEEE 802.11 standard defines an architecture for the Wireless Local Area Network that covers the physical and link levels present in the reference OSI communication model. For the physical level only, Radio Frequency (RF) and infrared (IR) transmissions are treated, but other forms of wireless communication such as microwave and visible light can also be considered. For the link level, the access control to the medium is addressed, through the definition of the MAC protocol (Medium access Control).

Taking into account the main characteristics of the IEEE 802.11 standard, such as interoperability, low cost, high market demand, reliability of project execution, there is a great growth in the use mainly of Local Area Networks of Wireless Computers, also known as Wireless Networks, in public and private environments. This makes Wireless Networks a priority resource in environments where it is most often possible to access the Internet, whether inside corporations, in homes or in public environments, such as shopping malls, airports and so on.

The architecture of Wireless Networks according to the IEEE 802.11 standard, is based on the division of the area covered by the Wireless Network into cells, these cells being called BSA (Basic Service Area). The size of the coverage of each BSA will depend exclusively on the characteristics of the environment itself and the power of transmitters and receivers used in the computational devices. The other components of the Wireless Networks architecture are listed below:

1) BSS (*Basic Service Set*): Which is the set of computational devices that communicate by broadcasting (BC) or infrared (IR) within a Basic Service Area;
2) AP (*Access Point*): Specific computational devices, which have the purpose of capturing the transmissions made by computational devices belonging to its BSA (Basic Service Area), which are destined to stations belonging to another Basic Service Area. The Access Point, in turn, will perform the retransmission using a distribution system;
3) Distribution System: Communication infrastructure, which has the purpose of performing the interconnection of several Basic Service Area to allow the construction of networks, which have covers larger than one cell;
4) ESA (*Extended Service Area*): Service Area that has the purpose of interconnecting several BSAs, through the Distribution System using the Access Point;
5) ESS (*Extended Service Set*): Which is intended to represent a set of computational devices consisting of the union of several BSSs (Basic Service Set) connected by a Distribution System.

The IEEE 802.11 standard also defines a medium access protocol, which is present in a MAC sublayer of the data link level. This protocol is called DFWMAC (*Distributed Foundation Wireless Medium Access Control*), which has two access methods, one of which is a distributed and mandatory feature. The other access method of the DFWMAC protocol is optional, having a centralized feature, and according to the IEEE standard, both the distributed method and the centralized method in the communication system can coexist. The medium access protocol also has the property of treating problems related to computational devices that try to move from one cell to another, a process called roaming. It is also related to the protocol of access to the medium of property to treat problems of lost computational devices, being able to be denominated of hidden node.

IV. Computational Intelligence Techniques

Computational Intelligence consists of an area of computing and engineering responsible for studying the computational principles that make intelligent behavior possible. Among the main techniques of this area are [5]:

- Artificial neural networks;
- Fuzzy Logic;
- Evolutionary Algorithms;
- Theory of Games

A. Artificial Neural Networks

The work related to Artificial Neural Networks, is inspired by the observation that the human brain has unusual computational properties. According to [6], the human brain represents a highly complex, non-linear and parallel information processing system.

An Artificial Neural Network is composed of relatively simple processing units, and the interconnections of these units are adapted from a learning algorithm.

Fig. 1 presents the neuron model present in the Neural Networks, which is composed of input signals, weights, sum function, transfer function and output.

These components are shown below:

- Inputs: It is the signal x_j present at the input of neuron k which is multiplied by the synaptic weight w_{kj}. In the synaptic weight the first index k refers to the neuron in question and the index j refers to the input terminal of the synapse to which the weight is referring.
- Sum function: It is intended to sum the input signals, taking into account the respective synapses of the neuron. The sum function can be obtained by Equation 1:

$$u_k = \Sigma\, w_{kj}\, x_j \quad (1)$$

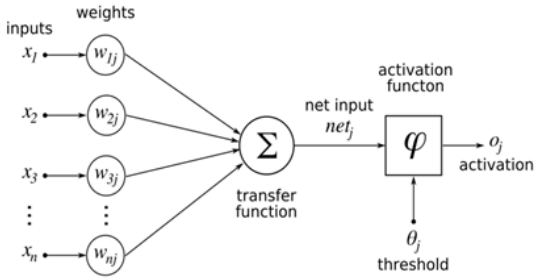

Fig. 1. Model neuron

- Activation Function: Also called "Transfer Function" that maps the sum to an end output.

The sigmoid activation function is the most used function in applications of Artificial Neural Networks, being composed by a S-shaped graph. The sigmoid function is represented in applications of Neural Networks by Equation 2:

$$\varphi(v) = \frac{1}{1+\exp(-av)} \quad (2)$$

In terms of its architecture, Artificial Neural Networks can be classified into two distinct groups: those that do not have recurrent connections, called acyclic ones, and the cyclical ones, which have it [7]. Direct feeding neural networks is organized in layers, and that certain layer can receive only inputs of neurons located in the layer immediately inferior or below. The inner layers (which do not connect to the outside world) are called hidden layers [6].

Acyclic Neural Networks of several layers have the fundamental characteristic of direct feeding, but with the presence of one or several layers hidden between the input layer and the output layer. The differentiation between Neural Networks that make use of hidden layers to those that do not use, for example the Perceptron Network, is the possibility of increasing the capacity of representation of transformations between inputs and outputs of the Neural Network.

B. Neural Networks Learning

The learning process consists in the adaptation of Synaptic values. For learning in Neural Networks there are several algorithms able to perform the adaptation of the parameters, so that after a finite number of iterations can converge to a viable solution.

For learning in Neural Networks there are several algorithms able to perform the adaptation of the parameters so that after a finite number of iterations can converge to a viable solution. The learning algorithm aims to reduce a cost function, usually associated with the error in the system output [6].

According to Rezende [8], the algorithms or learning techniques applied to Artificial Neural Networks can be classified according to three different principles:

- Supervised learning
- Non-supervised learning
- Reinforcement learning

Supervised learning has the purpose of enabling the learning of a given Artificial Neural Network through a set of input and output examples. Since the desired output is known for each example, it is possible to calculate the error and adjust the weights, in order to approximate the answer of the desired answer.

Unsupervised learning also has the purpose of enabling the learning of a particular Neural Network through the processing of a set of information, but without presence of a specialist (teacher), that is, for this type of learning it has not the knowledge of the desired outputs for the inputs entered during the training. The adjustments of the synaptic weights belonging to each entry are performed based on the input values [8].

Fig. 2 demonstrates the behavior of a neuron in the supervised learning process, having as fundamental elements the input vector, hidden neuron layer, output neuron, summation function. The neuron has an output $Y_k(n)$, which is the result of the activation function of the neuron. In Fig. 2 there is the presence of a desired output $d_k(n)$, which is possibly different from the output $Y_k(n)$ of the neuron. In this way the subtraction between the output generated by the neuron $Y_k(n)$ and the desired output $d_k(n)$ results in an error signal $e_k(n)$, which will be returned to the neuron, allowing the adjustment of its respective weights of the input layer, generating new outputs, as defined in the equations below:

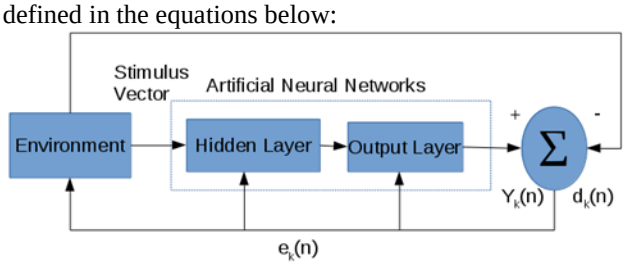

Fig. 2. Learning by error connection.

- The local gradient of the last layer is calculated through the error generated in the output layer and the derivative of the error using equation 3 [9].

$$\delta_j(n) = e_j(n)\varphi'_j(v_j(n)) \quad (3)$$

- Error Propagation: The local gradient of each neuron from the previous layers is calculated by equation 4 [9].

$$\delta_j(n) = \varphi'_j(v_j(n))\Sigma_k\delta_k(n)w_{kj}(n) \quad (4)$$

where:
j: index of the current layer neuron;
k: neuron index of the posterior layer.

- Synaptic weights: After calculating each local gradient, the adjustment of the synaptic weights is given by equation 5 [9].

$$\Delta w_{ji}(n) = \eta\varphi_j(n)y_j(n) \quad (5)$$

where:
η: network.

- The adaptation of the weights is performed by equation 6 [9].

$$w^{k+1}{}_{ji}(n) + \Delta w_{ji}(n) \quad (6)$$

- For each pattern presented to the network, the instantaneous error in equation 7 [9] is measured.

$$\varepsilon(n) = 1/2\Sigma_{j \in C}\, ej^2(n) \quad (7)$$

where:
C: set of all neurons in the output layer

C. Multilayer Perceptron Neural Networks

The Multilayer Perceptron is an Artificial Neural Network architecture composed of an input layer, an output layer and at least one hidden layer between input and output. This type of Neural Network is used in a large scale to solve complex problems, as it has as a characteristic the supervised learning, through the use of the backpropagation error or backpropagation algorithm [6].

The backpropagation algorithm, also known as backpropagation, is used in artificial neural networks with supervised architecture to adjust the synaptic weights, minimizing the errors between the output generated by the neurons and the desired output, that is, to reduce the error rate in each learning cycle [10].

D. Self-Organizing Maps

Artificial Neural Networks called Self-Organizing Maps (*SOM*) are systems that are organized internally to represent the distribution of input data, without the presence of a supervisor. The Kohonem Networks are inspired by the fact that information in the human brain is spatially organized. Areas of the cortex may be said to form "maps" of sensory spaces relating to neurons responsible for specific responses to certain stimuli in certain regions, such as specific responses to frequencies in the areas of the brain intended for hearing and vision. In this sense the Kohonen Networks aim to generate agglomerates with high response activity to a given stimulus.

Self-Organizing Maps (SOM) consist of two layers: The input layer I and the output layer U. The input of the Kohonen Network is a vector in the d-dimensional space in R^d, defined by $x_k = [\xi_1,...,\xi_d]^T$, K = 1, ..., n; each neuron j has a vector w, also represented in the space R^d associated with the input vector x_k, $w_j = [w_{j1}, ..., w_{jd}]^T$. Neurons are interconnected through a neighborhood relation shown in the map structure.

From this, taking into account the state of activation of neuron *i* of the Kohonen Map in relation to the stimulus of the input vector x_k:

1. $y(x_k) = 1$, if $i(x_k) = \arg_j \min\|x_k - w_j\|$;
2. $y(x_k) = 0$, otherwise.

Where:

- $i(x_k)$: i, j represent specific neurons in the network;
- $\|.\|$: It is the distance measure, through the Euclidean norm, $1 \geq j \geq N$. Being N the number of neurons in the output layer;
- $y(x_k)$: Informs the state of activation of the position *i* of the Kohonen Map in relation to the stimulus of the input vector x_k.

The neurons receive the same value the neuron whose weight vector is closer to the input vector get activated. For the winning neuron will be assigned a certain neighborhood. Neurons in this neighborhood will have the opportunity learning, by adapting their weights following Equation 8.

$$w[t + 1] = w[t] + \alpha[t](x[t] - w[t]) \quad (8)$$

where:

- w [t + 1]: Is the value of the weight updated;
- α [t]: It is the learning constant.

Another characteristic in the process of neighborhood determination is to ensure a similarity between the neurons that are part of a given region, this region being composed of the winning neuron and its neighbors. According to Kohonen [11], in order to guarantee similarity, one must apply the adjustments of the weights, as presented in Equation 8, both in the winning neuron, both in its neighborhood. In this way it allows the map to be organized geographically, because the neurons that do not belong to the neighborhood will not have weights adjusted.

The neighborhood concept ensures that close neurons respond to similar patterns, thus creating a self-organizing feature map.

E. K-Means Algorithm

The K-Means algorithm, is a simple technique that can be used to analyze groups. This algorithm is proposed by Macqueen in 1967. The algorithm is applied when grouping certain objects into groups called clusters. According to Macqueen [12] these groups or clusters are

formed through the application of distance measurement techniques or similarity techniques between objects.

K-Means uses a partitioning technique, in which the grouping is performed through optimization through the application of an objective function. This objective function is based on prototypes that have the principle of finding *n* clusters *k*, being the value of n determined by the user [13].

The non-hierarchical grouping process therefore defines a number *k* of classes and also performs an initial classification of n objects in *k* classes, being the value of *k* determined by the user before or after the grouping process.

In terms of programming and computational processing, the K-Means algorithm is easy to program and economical, not requiring high computational power. The K-Means algorithm is able to process large volumes of data, the storage complexity of which is O((m + K) n), where m is the number of points and n is the number of attributes [12].

However, the K-Means algorithm has some disadvantages that need to be analyzed beforehand which are:

- In a large database, the K-Mean algorithm cannot be efficient in generating quality solutions if its initialization is not successful, as well as its initial centroids representing the groups being poorly positioned in the search spacing;
- In terms of performance, the algorithm does not guarantee the optimum overall result, since the final quality of the solution depends on the initial sets of clusters, and can remove them from the overall optimum result;
- Inappropriate choice of the value of k can result in poor results.

The K-Means algorithm therefore has the purpose of converging to a solution using combinations of proximity functions as well as types of centroids reaching a state in which no point, or data object, changes group, consequently there will be no change of centroid. In some cases, the algorithm may not achieve these aims, and it is necessary to assign a weaker condition to reach the final state, such as repeating this process until only 1% of the objects do not change groups.

V. Methodology

This work aims to apply Computational Intelligence techniques to the problem of identifying anomalies in wireless network traffic, more specifically, neural networks and k-means algorithm. As mentioned in the previous sections, the techniques adopted for this work are: Artificial Neural Networks, more precisely with the Multilayer Perceptron algorithm and the Self-Organizing Map (SOM) algorithm, and the K-Means algorithm.

In order to achieve the proposed aims, the following activities were performed in accordance with the chronological order of execution.

We use a database with examples of specific anomalies in wireless networks.. This base in turn is the final product of the work entitled *A Methodology for building a Dataset to Assess Intrusion Detection Systems in Wireless Networks* [1].

The next step is to perform a pre-processing in the database, in such a way that two new databases are obtained. One of the databases is composed of only 10% of the data from the original database and is destined for the test step in the selected algorithms. The other database is composed of 90% of the data from the original database and is destined for the training step of the selected algorithms. Both databases are stored in the Database Manager System named PostgreSQL, and are accessed by the Weka software (*Waikato Environment for Knowledge Analysis*).

Finally, the results of each selected algorithm are analyzed and formatted through tables. In relation to the results, the following information is presented for analysis: Percentage of Classification, labeling of groups, relation of correctness and errors.

VI. Case Study

The case study chosen to analyze the results of the application of computational intelligence techniques presented in previous sections uses data from real wireless networks [1] and the data mining software, Weka [14].

A. Database

The database defined for the execution of this case study is a real collection of network traffic captured in the Wireless architecture. This data, in turn, is obtained by the behavior of users to access various information as well as for the use of the Internet. According to the authors [1], the network traffic obtained by students and employees of the institution in which the experiment was performed was used for this database.

The database chosen for the experimentation of this work made use of two different scenarios. The scenarios discussed have their own configuration and topologies, being a scenario of home environment typical of wireless networks, while the other is a more complex environment, being a corporate environment.

This database is composed of a total of 616,047 records, each record being composed of 16 variables that are characteristics of the wireless network traffic itself. Also in each record of the database is defined a last variable the class to which belongs certain registry, classification is realized taking into account the values of the sixteen variables referring to the obtained wireless network traffic. In this way the data are classified in:

- *Normal*: Acceptable wireless network traffic;
- *EAPOLStart*: Traffic using the Extensible Authentication Protocol (EAP), which aims to perform an authentication method in both the Wired Equivalent Providence (WEP) protocol, both Wi-Fi

Protected Access (WPA) protocol, commercial versions for wireless network access;

- *Beacon Flood*: Management type requests, which are intended to transmit millions of invalid Beacons, resulting in the difficulty that a certain Wireless network device will have in identifying a legitimate Access Point [15];
- *Deauthentication*: It also represents management-type requests, which are injected from the Wireless Network. The frames belonging to this anomaly are transmitted as fictitious requests, which requests the deactivation of a device that is authorized in the Wireless Network;
- *RTSFlood*: Also called Request-to-Send Flood is a control-type frame. This anomaly is based on the large-scale transmission of RTS frames or frames for a short period of time [15].

The database for the experimentation process of this work is divided into two distinct bases, in order to meet the requirements of each defined intelligent algorithm. In this way a training database is generated respecting the characteristics of each algorithm, being composed by 554,442 registers, which corresponds to 90% of the complete database. Also, the test database is generated, being composed by 61,604 records that correspond to 10% of the complete database, respecting the characteristic of each algorithm. In order to optimize the experimentation process and to provide better data manipulation, the training and test databases for each defined computational intelligence technique are stored in the PostgreSQL Database Management System.

B. Experiment 1 – Multilayer Perceptron Algorithm

The Multilayer Perceptron algorithm is one of the algorithms of the Artificial Neural Network that has the characteristic of being supervised, that is, it requires the presence of a specialist in the learning process.

For the realization of the experiment, the Weka software [14] is used through the classification process. In this classification step, one must select the classification algorithm called Multilayer Perceptron, which is assigned properties for its execution:

- Hidden layers of the network are used;
- Learning rate does not decrease with the growth of number of times;
- The number of times to train through.

In order to perform the tests of the Multilayer Perceptron algorithm, the training and testing databases are stored in the PostgreSQL software database. The validation of the algorithm is performed through the use of Cross-Validation, Percentage Separation and Testing techniques.

The cross-validation technique has the characteristic of dividing the database into 10 subsets, in which 9 sets are used for training and one for evaluation. In order to perform this evaluation of the Multilayer Perceptron algorithm we use the complete database, which is composed by 616,047 records.

Percentage separation or Percentage split is a test procedure that has the characteristic of using 66% of the training base, the rest being used for the tests. Also in this case, the complete database (616,047 records) is used for the validation of the Multilayer Perceptron algorithm.

The supplied test set or Supplied test set is a test procedure that makes use of two distinct databases, one for supervised learning of the artificial neural network, while the other database is intended for testing.

Table I presents the percentage of detection of the classes for each of the test procedures performed, while Table II presents the percentage of Registers classified correctly and incorrectly by the Multilayer Perceptron algorithm for each executed test procedure.

TABLE I: ACCURACY – MULTI-LAYER PERCEPTRON

	Cross-Validation	Percentage Split	Supplied Test
Normal	84,97%	87,31%	96,25%
EAPOLStart	4,17%	4,30%	0,81%
BeaconFlood	1,58%	0,93%	0,92%
Deauthentication	2,61%	2,54%	26,65%
RTSFlood	0,02%	0,02%	40%

TABLE II: PERCENTAGE OF ERRORS – MULTI-LAYER PERCEPTRON

	Accuracy	Errors
Cross-Validation	93,34%	6,65%
Percentage Split	95,10%	4,89%
Supplied Test	95,53%	4,46%

C. Experiment 2 – Self-Organizing Maps

Self-Organizing Maps, also called Kohonen Maps is a type of Artificial Neural Networks that have as fundamental principle the competitive procedure of learning between the units of the network.

The main purpose of Kohonen Maps is the possibility of building systems that are organized internally through the distribution of incoming data without the presence of a particular expert. In this sense, Self-Organizing Maps will present in the output the formation of settlements, also called clusters, which have a maximum response to a given stimulus.

In this classification step, it must select the classification algorithm called SOM, which is assigned properties for its execution:

- Initialization of the input vector;
- Define the learning function during training;
- Define the neighborhood function;
- Initialize the initial size of the neighborhood;
- Define the number of training interactions.

In order to perform the tests of the SOM algorithm, the same test options assigned to the Multilayer Perceptron algorithm experiment will be used, which are: Cross-validation Folds 10, Percentage split 66% and Supplied Test Set. In order to perform the Cross-Validation and Percentage Split tests, the complete database (616,047

records) is used, whereas for the Supplied Test Set option, a training database (90% of the complete database) and a base (10% of the complete database).

Table III presents the percentage of detection of the classes for each of the test procedures performed, while Table IV presents the percentage of Records classified correctly and incorrectly by the SOM algorithm for each test procedure performed.

TABLE III: PERCENTAGE OF CLASSIFICATION – SOM

	Cross-Validation	Percentage Split	Supplied Test
Normal	88,07%	88%	88,04%
EAPOLStart	0,59%	0,54%	1,04%
BeaconFlood	0%	0%	0%
Deauthentication	0%	0%	0%
RTSFlood	0%	0%	0%

TABLE IV: PERCENTAGE OF ERRORS – SOM

	Accuracy	Error
Cross-Validation	88,65%	11,34%
Percentage Split	88,54%	11,45%
Supplied Test	89,08%	10,91%

D. Experiment 3 – K-Means Algorithm

The K-Means algorithm is a technique that uses K-Mean data clustering, also called *K-means clustering*. This algorithm seeks to find the best division of data into *K* groups C_i, where i = 1, 2, 3, ..., *K*. Thus, we obtain that the total distance between the data of a given group and its respective center is minimized.

The K-Means algorithm follows the steps below:

- At this step, each point represented by a given P is shifted to its respective group, which corresponds to the nearest mean vector;
- The algorithm calculates again the means of the vectors and also performs the distribution of the data in each group;
- This process of reallocating data to new groups, on which the mean vectors are the closest, is performed until all data are in its groups.

In this step of clustering, it must use the algorithm called Simple K-Means, which is assigned properties for its execution:

- Distance calculation function used, such as Euclidean Distance;
- Define the number of iterations of the algorithm;
- Define the ideal number of clusters to be used to achieve a good result.

To perform the K-Means algorithm tests, the following test options will be used: Supplied Test Set. For the Supplied Test Set, a training database (90% of the complete database) and a test database (10% of the complete database) are used. However, because the K-Means algorithm is unsupervised, data pertaining to the class label is not used for training.

A total of 500 iterations and 25 clusters or groups are defined for the K-Means algorithm. Also, the accuracy for each predefined class (Normal, EAPOLStart, Beacon Flood, Deauthentication, RTS-Flood) is calculated, and these data are presented in Table VI.

TABLE V: CLUSTERING – K-MEANS

	Normal	EAPOL Start	Beacon Flood	Deauthentication	RTSFlood
Clusters	22	2	0	1	0

TABLE VI: PERCENTAGE OF ACCURACY AND ERRORS – K-MEANS

	Accuracy	Errors
Normal	98,39%	1,61%
EAPOLStart	78,48%	21,52%
Beacon Flood	0%	100%
Deauthentication	91,76%	8,24%
RTSFlood	0%	100%

Results show that, with the chosen parameters, the k-means algorithm could not identify all predetermined classes.

For applications of Wireless Networks, which requires an identification of anomalies preferably in real time, as well as the frequent practicality of these anomalies, this algorithm partially meets the desired objective for this context [16]. In order to improve the results of the K-Means algorithm, a technique called Variable Selection, also known as Select Feature, is applied.

The technique of selecting variables has as main objective to select the attributes or variables that can effectively contribute to the achievement of a certain algorithm, that is, it will eliminate attributes considered redundant or irrelevant [2].

In order to improve the performance, as well as the own results obtained by the K-Means algorithm, the CfsSubsetEval algorithm for the identification of the evaluator attribute is applied and for the search method the BestFirst algorithm is used. The first algorithm aims to evaluate a subset of variables considering the individual predictability of each resource, as well as the degree of redundancy among them [17]. The second algorithm aims to searching in a space of attributes subsets that are closer to the objective, which will lead to reach the optimal state more quickly [18].

After applying the technique of selection of variables to the database, which contains 17 attributes, are selected two attributes, which are identified by the third and fifteenth attributes, that is, making use of these two attributes it is possible to achieve the objective more quickly and efficiently.

To perform the test using only the two attributes selected, a total of 500 iterations and 25 clusters or groups were defined for the K-Means algorithm. Fourteen *clusters* identify the Normal class, 3 *clusters* identify the EAPOL Start class, 6 *clusters* identify the Beacon Flood class, 1 *cluster* identifies the Deauthentication class, and

no *cluster* identifies the RTSFlood class. The percentage of correctness and errors for each pre-defined class (Normal, EAPOLStart, Beacon Flood, Deauthentication, RTSFlood) is also performed. These data are presented in Tables VII.

TABLE VII: PERCENTAGE OF ACCURACY AND ERRORS – K-MEANS

	Accuracy	Errors
Normal	97,92%	2,08%
EAPOLStart	59,99%	40,01%
Beacon Flood	68,31%	31,69%
Deauthentication	91,95%	8,05%
RTSFlood	0%	100%

VI. DISCUSSION OF THE RESULTS

The results obtained in the application of the tests in the three experiments described in section 6 point to good results in the application of classification techniques, also known as pattern recognition, for data from wireless networks. Table II, Table IV and Table VI represent the percentage of correctness and errors in relation to the database for the *Multilayer Perceptron* classification algorithm, the *Self-Organizing Maps* algorithm with the Learning Vector Quantization (LVQ) method and the unsupervised learning algorithm K-Means respectively.

The algorithm that best meets the identification needs of the classes determined in the data base of this work is the Perceptron Multilayer classification algorithm with the use of backpropagation training. This algorithm is validated by experiment 1, with 95.53% of the records present in the database being correctly identified for the test set. The other two types of tests adopted (Cross-Validation and Percentage Split) also present satisfactory classification results, with 93.34% and 95.10% respectively.

The *Self-Organizing Maps* and *K-Means* algorithms partially attend the problem of identification of the classes adopted in the database. The first one, according to Table III and Table IV, presents correct identification in 89.08% of the registers, with only the *Normal and EAPOLStart* classes being identified, making it possible to use intrusion detection in wireless networks together with other techniques. The second algorithm presents correct identification in 94.69% of the registers, identifying the *Normal, EAPOLStart, Beacon Flood, Deauthentication* classes, and being unable to identify only the class *RTSFlood*. Although the Multilayer Perceptron algorithm is recommended as the algorithm that best meets the identification needs of the determined classes, the K-Means algorithm, after applying the variable selection technique, achieves satisfactory results, considering the organization of the data and its quantization.

VII. CONCLUSION AND FUTURE WORKS

The aim of this work was to evaluate some computational intelligence algorithms capable of identifying or classifying some anomalies found in the wireless networks traffic. These algorithms are able to minimize the difficulties that managers have in controlling the various members of these networks, as well as in real-time identification of various anomalies.

The algorithms present in this work represent classification techniques, which have the characteristic of supervised and unsupervised learning. For the evaluation of supervised classification, the Multilayer Perceptron algorithm with backpropagation learning is used. For the evaluation of the unsupervised classification, also called the clustering process in which no specialist is present, the K-Means algorithm and the algorithm of the Self-Organizing Maps with Learning Vector Quantization (LVQ) method are used.

The validation process of the selected algorithms involves the use of a real database of Wireless Network traffic, which has the following anomalies: EAPOLStart, Beacon Flood, Deauthentication and RTSFlood. This database is composed of 17 variables per record, 16 attributes of the MAC layer and an attribute of identification of the class to which a particular record belongs.

The best result for the classification of anomalies in wireless environments is the Multilayer Perceptron classification algorithm, since it presents a general correct classification rate of 95.53% and 93.34% for the cross-validation test of the collected database and also classifies all predefined classes (Normal, EAPOLStart, Beacon Flood, Deauthentication, RTSFlood). The other algorithms can also obtain a high percentage of correct answers in the classification, but they cannot identify all the predefined classes. Although the *K-Means* algorithm is partially attend the problem of identification of the classes adopted in the database, it is necessary to apply a pre-processing of variable selection, which can be useful when there is no supervised data.

Future work with the intention of continuing the proposal in this article involves the evaluation of other algorithms of pattern recognition with supervised and unsupervised learning techniques. Also, as future work we can verify the recommendation of supervised or unsupervised classification algorithms with wireless network traffic using WPA (*Wi-Fi Protected Access*) and WEP (*Wired Equivalent Privacy*).

REFERENCES

[1] E. W. T. Ferreira, *et al*., "A methodology for building a dataset to assess intrusion detection systems in wireless networks," *WSEAS Transactions on Communications*, vol. 14, pp. 113–120, 2015.

[2] G. C. F. Sahin, "A survey on feature selection methods," *Computers Electrical Engineering*, 2014, pp. 16–28.
[3] M. Govindarajan and R. M. Chandrasekaran, "Intrusion detection using neural based hybrid classification methods," *Computer Networks* vol. 55, no. 8, pp. 1662–1671, 2011.
[4] E. W. T. Ferreira, *et al.*, "Intrusion detection system with wavelet and neural artifical network approach for networks computers," *IEEE Latin America Transactions*, vol. 9, no. 5, pp. 832–837, 2011.
[5] S. Shamshirband, *et al.*, "An appraisal and esign of a ulti-agent system based cooperative wireless intrusion detection computational intelligence technique," *Engineering Applications of Artificial Intelligence*, vol. 26, pp. 2015–2127, 2013.
[6] S. Haykin, *Neural Networks and Learning Machines*, 3nd. Ontario Canada: Pearson, 2009.
[7] P. N. E. S. J. Russel, *Inteligência Artificial*, 2nd. Rio de Janeiro: Elsevier, 2004.
[8] S. O. Rezende, *Sistemas Inteligentes Fundamentos e Aplicaõess*, 1nd. Barueri - SP - Brasil: Editora Manole, 2003.
[9] M. F. Alves and A. D. P. L. M. L. M. Lopes, "Selecão de variáveis stepwise aplicadas em redes neurais artificiais para previsão de demanda de cargas elétricas," *Simpósio Brasileiro de Automacão Inteligente*, 2013.
[10] M. Ibnkahla, "Applications of neural networks to digital communications: A survey," *Signal Processing - Special Issue on Emerging Techniques for Communication Terminals*, vol. 80, no. 7, pp. 1185–1215, 2000.
[11] T. Kohonen, *Self-Organizing Maps*, 3nd. Springer-Verlag Berlin Heidelberg, 2001.
[12] J. B. Macqueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, 1967.
[13] C. P. P. Furlan, "Análise da Rede Social Tocantins Digital, utilizando o Algoritmo k-médias e centralidade de intermediação," Diss. de mestrado. Pontifícia Universidade Católica de Goiás, 2014.
[14] M. Hall, *et al.*, "The WEKA data mining software: An update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.
[15] R. F. de Moraes and N. V. D. S. A. C. Maciel, "Avaliação de um conjunto de dados quanto á sua qualidade na especificação de perfis de ataque e não-ataque numa rede IEEE 802.11w," *Anais da VI Escola Regional de Informática da Sociedade Brasileira da Computação(SBC) - Regional de Mato Grosso*, 2015, pp. 145–1508.
[16] A. Dorri and S. R. K. E. Kheirkhah, "Security challenges in mobile ad hoc networks: A survey," *IJCSES*, vol. 6, no. 1, 2015.
[17] M. A. Hall, "Correlation-based feature subset selection for machine learning," Tese de doutorado. Hamilton, New Zealand: University of Waikato, 1998.
[18] R. E. K. D. M. Chickering, "Best-first minimax search," *Atificial Intelligence*, vol. 84, pp. 299–337, 1996.

Daniel R. Canêdo has a degree in Computer Engineering from Pontifícia Universidade Católica de Goiás (2003) and a Master's degree in Electrical Engineering from the University of Brasília (2006). He is currently an exclusive professor of the Federal Institute of Goiás - Campus Luziânia. He is currently a PhD student in the Post-Graduate Program in Electronic Systems and Automation Engineering of the Department of Electrical Engineering of the University of Brasília (UnB).

Alexandre R. Romariz holds a BS in Electrical Engineering from the University of Brasília (1992), a Master's degree in Electrical Engineering from the State University of Campinas (1995) and a PhD in Electrical Engineering from the University of Colorado at Boulder (2003). He is currently "Professor Associado" at the University of Brasilia. He has experience in Computational Intelligence, Integrated Circuits, Optoelectronics and Digital Signal Processing.