Multiclass decomposition and Artificial Neural Networks for intrusion detection and identification in Internet of Things environments

Cristiano Antonio de Souza[1], **João Vitor Cardoso**[1], **Carlos Becker Westphall**[1]

[1]Departamento de Informática – Universidade Federal de Santa Catarina (UFSC)
Caixa Postal 476 – CEP 88040-900 – Florianópolis – SC – Brazil

cristiano.souza.c@posgrad.ufsc.br, joao.v.c@grad.ufsc.br

westphal@inf.ufsc.br

Abstract. *The Internet of Things (IoT) systems have limited resources, making it difficult to implement some security mechanisms. It is important to detect attacks against these environments and identify their type. However, existing multi-class detection approaches present difficulties related to false positives and detection of less common attacks. Thus, this work proposes an approach with a two-stage analysis architecture based on One-Vs-All (OVA) and Artificial Neural Networks (ANN) to detect and identify intrusions in fog and IoT computing environments. The results of experiments with the Bot-IoT dataset demonstrate that the approach achieved promising results and reduced the number of false positives compared to state-of-the-art approaches and machine learning techniques.*

1. Introduction

Internet of Things (IoT) is a paradigm that integrates a set of physical objects, such as sensors, actuators, objects, and applications through a network for interaction and cooperation [Conti et al. 2018, Oppitz and Tomsu 2018]. Currently, there are several electronic devices and household appliances that can be connected to the Internet, making data and remote services available to users [Vikram et al. 2017]. Smart sockets and light bulbs that can connect to the wireless network, allowing it to be controlled remotely, are examples of these devices. In addition, IoT is expanding into various application areas such as residential and industrial automation. IoT environments are composed of devices with constrained resources, low processing power, and limited memory. These devices usually have sensors that collect information from the environment, generating a large amount of data. To carry out the storage, processing, and analysis of the collected information, devices with greater computational power are needed. Sending this data for processing in cloud computing can cause latency issues due to the large volume of data generated by these devices and the distance between servers and IoT devices. Fog Computing allows data processing closer to IoT devices, acting as an intermediary layer between these devices and the cloud [Iorga et al. 2018].

The resource constraints lead to difficulties in the implementation of security mechanisms in IoT devices [Frustaci et al. 2017]. Security breaches are common in IoT systems because, in many cases, security issues are neglected due to resource constraints such environments and the cost of implementing security mechanisms. The Internet of Things is increasingly present in people's daily lives through everyday devices. However,

it still has many vulnerabilities. In 2016, many IoT devices were invaded through Mirai malware and used to make denial of service attacks, impairing some functionality of services such as Twitter and Spotify, making them unavailable for a few hours.

Intrusion detection is one of the key security points, aiming to identify attack attempts by malicious entities. Network-based detection approaches analyze the traffic that travels through the network without interfering with communication between devices to find out if an attack is being carried out [Garcia-Teodoro et al. 2009]. In addition to detecting the occurrence of the attack, it is also important to identify the type of attack that is taking place to carry out specialized countermeasures for each type of attack and assist in decision-making by the person responsible for the network. According to the authors Hughes, McLaughlin and Sezer [Hughes et al. 2020] Intrusion Response Systems, which aim to respond to attacks automatically as soon as they are alerted, need specific knowledge to apply precise and effective countermeasures. However, it is more difficult to obtain a high success rate in the discrimination of different classes. Some approaches found in state of the art presented difficulties related to false positives [Diro and Chilamkurti 2018, Bhuvaneswari and Selvakumar 2020, Shafiq et al. 2021] and others the detection of some types of attacks [Almiani et al. 2020, Shafiq et al. 2021]. These difficulties are mainly related to discriminating traffic into different classes is a more complex problem. Furthermore, the nature of the IoT environment necessitates less robust approaches, which makes this multi-class detection difficult. This work advances state of the art in this regard, bringing as a contribution a two-stage detection architecture that allows the release of benign traffic quickly and performs deeper analysis in the non-benign traffic. The architecture involves deep artificial neural networks, decomposition technique, attribute selection, and class balancing, seeking to reduce false positives and improve the identification of attacks.

The general objective of this work is to propose a detection approach based on Deep Neural Networks (DNN) Perceptron MultiLayer and in multiclass problem decomposition strategy One vs. All (OVA) for intrusion detection in the context of fog computing and the Internet of Things. The results obtained through experiments with the BoT-IoT intrusion dataset demonstrate that the approach has achieved promising results compared to machine learning methods and reduced the number of false positives compared to related work approaches.

The rest of the work is organized as follows. Section 2 presents related works found in the state of the art. Section 3 presents the proposed multi-class detection approach. Section 4 describes the experiments performed and discusses the results obtained. Finally, Section 5 presents the conclusions of the work.

2. Related works

This section presents intrusion detection and identification solutions proposed for environments based on fog computing and IoT. A review of the literature on the subject was carried out. Articles were searched in the IEEE, ACM Digital Library, Elsevier, and Springer databases. Table 1 presents a comparison between works found in state of the art. It is possible to identify the techniques used and the dataset used for validation.

Almiani et al. [Almiani et al. 2020] proposed an approach that performs event classification using deep neural models, in this case investigating the use of Deep Re-

Table 1. Related works.

Work	Method	Validation	Observations
[Diro and Chilamkurti 2018]	DNN	NSL-KDD	Difficulties related to false positives
[Abdel-Basset et al. 2020]	RNN	Bot-IoT	Distributed learning architecture
[Bhuvaneswari et al. 2020]	VCDL	Bot-IoT	Difficulties related to false positives
[Soe et al. 2020]	CST-GR + DT	Bot-IoT	No detection rates for benign traffic
[Almiani et al. 2020]	RNN	NSL-KDD	Difficulty in detecting some attacks
[Popoola et al. 2021]	BLSTM	Bot-IoT	Difficulties related to false positives
[Shafiq et al. 2021]	CorrAUC + RF	Bot-IoT	Difficulties related to false positives
[Du et al. 2020]	SVM+PCA	NSL-KDD	Difficulty in detecting some attacks
Our work	**DNN+OVA(DNN)**	**Bot-IoT**	No countermeasures

current Neural Network (DRNN). In the neural model training process, oversampling balances the database and avoids discriminatory behavior for classes with more meaningful examples. The goal is to improve the accuracy of attacks that have fewer instances of training. These approaches carried out the validation of the solutions with the NSL-KDD database, an old database. Furthermore, they presented some difficulties in detecting some types of attacks. The authors Abdel-Basset et al. [Abdel-Basset et al. 2020] also worked with recurrent networks. However, in this case, using the Local Gated Recurrent Unit (LocalGRU) to learn local representations. Furthermore, they introduced the Multi-head Attention (MHA) layer to capture and learn the global representation. This approach presents an interesting distributed learning architecture in fog. Du et al. [Du et al. 2020] presented a Principal Component Analysis (PCA) based approach to reduce the dimensionality of the data, eliminate the correlation between attributes and reduce the training time of the SVM classifier for fog intrusion detection. These works presented difficulties in detecting some types of attacks, such as privilege escalation.

Diro and Chilamkurti [Diro and Chilamkurti 2018] proposed a distributed approach based on Deep Neural Networks to detect intrusions in an IoT environment. The detection approach is distributed among the fog layer nodes. Each node has a DNN detection model, and the models' training is carried out between the distributed nodes. The approach was validated only with the NSL-KDD dataset. Bhuvaneswari and Selvakumar [Bhuvaneswari and Selvakumar 2020] proposed an algorithm to detect anomalies in IoT traffic using Vector Convolutional Deep Learning (VCDL) [Bhuvaneswari Amma and Subramanian 2018], a variation of Convolutional Neural Networks (CNN). The term convolutional vector is used because the proposed approach works with a vector and not a matrix. To validate the approach, the Bot-IoT dataset was used. Authors Popoola et al. [Popoola et al. 2021] proposed to reduce the dimensionality of the large-scale IoT network traffic data resource using the short-term long-memory (LAE) AutoEncoder encoding phase. In order to classify the network traffic samples correctly, the long-term interrelated changes in the low-dimensional resource set produced by LAE are analyzed using in-depth Short-Term Long Bidirectional Memory (BLSTM). The experiments were performed with the BoT-IoT dataset, and the results show that LAE significantly reduced the memory required for large-scale network traffic storage. The approach showed 92% benign traffic identification, which may indicate an increase in false positives. Soe et al. [Soe et al. 2020] proposed a lightweight machine learning-based approach using a new feature selection algorithm named Gain Ratio Correlated Set Threshold (CST-GR) and Decision Tree (DT), designed and implemented in the Raspberry Pi.

The approach is validated against the current Bot-IoT dataset but does not provide results on benign traffic classification. The authors Shafiq et al. [Shafiq et al. 2021] proposed a new feature selection approach called CorrAUC. The new algorithm relies on the wrapper technique to accurately filter resources and select effective resources for the machine learning technique using the Area Under the Curve (AUC) metric. The proposed approach is validated with the Bot-IoT dataset and four different ML algorithms, including Random Forest (RF), which obtained the best performance. These approaches presented difficulties regarding false positives. The obtained normal traffic identification rates were lower than expected. Thus a large amount of benign traffic may be mistakenly blocked. This can be very detrimental to the functioning of the network.

Multiclass detection is important to provide more accurate information for the engine to mitigate intrusions. In addition, this information can be used for decision-making by the person responsible for the network structure. However, state-of-the-art works present some difficulties in multiclass detection because it is a more complex problem to discriminate in different classes and because they seek less robust detection approaches to operate in this IoT and fog computing environment. Some approaches had weaknesses related to false positives and others the detection of some types of attacks. Thus, in this work, we present a two-stage analysis architecture, allowing quick detection of benign traffic and more robust analysis of traffic detected as intrusive. The objective is to employ a more robust method, such as the One vs. All (OVA) decomposition strategy, only in the second stage to discriminate intrusive traffic into categories. This technique decomposes the multiclass task into several smaller binary tasks and combines them into a final result. This approach is expected to be able to obtain a more accurate identification of intrusive traffic into categories. This second stage can even be deployed on a device with greater computing power, such as cloud computing.

3. Proposed Approach

Considering the growing popularization of IoT systems, their deficiencies about security, and related research on methods for intrusion detection and identification in these environments, this work proposes an approach based on decomposition strategy One-Vs-All (OVA) and Artificial Neural Networks (ANN) Multilayer Perceptron (MLP) to improve intrusion detection and identification performance. As can be seen in Figure 1 the work considers the context of an IoT system based on fog computing.

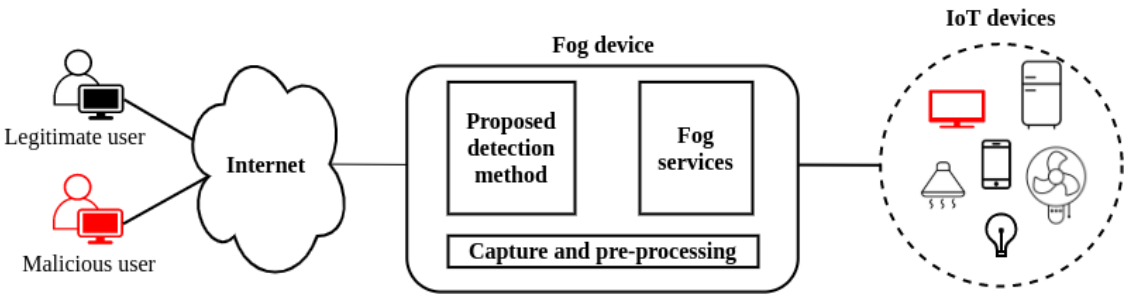

Figure 1. Architecture of the proposed approach.

The fog device acts as an intermediary layer between the IoT devices and the Internet. In addition, it processes data generated by IoT devices and provides services to

the network and users. In this context, malicious users can try to invade or corrupt the network of the IoT environment to stop functioning and steal data from the devices. It is observed that the proposed approach is embedded in the fog device. At that point, it can monitor the entire network by capturing the traffic generated by the devices. Therefore, this work considers that the device has mechanisms to capture, pre-process and format the traffic coming from the network. In this way, traffic can be submitted to the proposed detection approach to analyze and verify whether a given flow corresponds to a type of intrusion attack. The main aspects of the proposed approach are presented below.

3.1. Detection approach

The proposed detection approach has two detection stages. Figure 2 presents an outline of the proposed approach. First, a binary classifier based on the Deep Neural Network MultiLayer Perceptron distinguishes between benign and non-benign traffic. Performing this initial binary detection allows an immediate release of benign traffic and a more robust analysis of intrusive traffic since, as it has already been detected as intrusive, there is no urgency in response time. Although not considered in this work, the presented architecture allows this second stage of detection to be implemented in a device with greater computational power, such as cloud computing, as the non-requirement of real-time in this identification circumvents the latency problem the existing in the Fog communication with data centers.

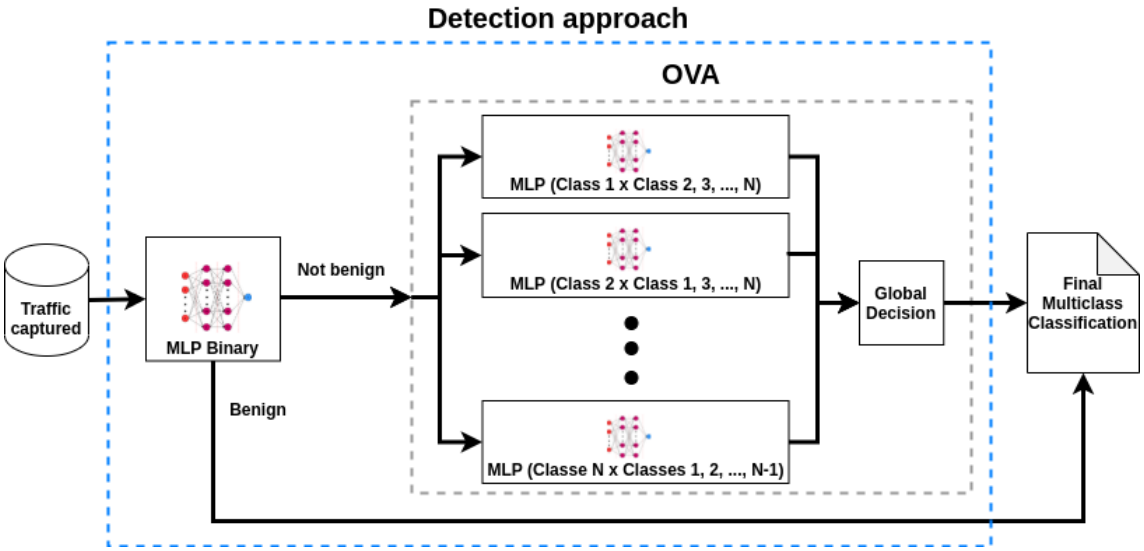

Figure 2. Proposed detection method based on the One-Vs-All strategy and DNNs.

Thus, the intrusive events are submitted to the classifier based on the multiclass problem decomposition strategy One vs. All (OVA) to identify the type of intrusion. This strategy creates a robust classifier with a higher computational cost, composed of several binary classifiers. Where K base classifiers are used for the existing K classes, where each classifier is modeled to discriminate a respective class [Oong and Isa 2012]. The base classifiers used together with the OVA strategy in the second detection stage are also DNNs. Finally, the outputs of the base classifiers are combined through a global decision that will indicate the final multiclass decision for an input data [Lorena et al. 2008]. It is

expected that this architecture using the robust classifier can increase and provide stability to the detection rate metric of all analyzed attacks.

As can be seen, the proposed detection method is based on MultiLayer Perceptrons neural network classifiers. These networks are based on the neural structure of the human brain. They are used to solve problems in several areas, such as optimization, linear and non-linear programming, pattern recognition, and computer vision [Chua and Yang 1988]. A big advantage of using these models is the ability to deal with complex natural systems that have large amounts of information [Abiodun et al. 2018]. MLP Feed-Forward neural networks are organized in layers, where the neurons of the previous layer are connected to the neurons of the next layer. Feed-Forward neural networks do not have connections between neurons of the same layer and do not have cycles in their connection graphs. These networks have an input layer and an output layer. Also, they can have several hidden layers located between the input and output layers. Networks with more than one hidden layer are known as Deep Neural Networks [Haykin et al. 2009]. Figure 3 shows the base architecture of the DNNs used as base classifiers of the proposed OVA approach. The input layer and its two hidden layers have n neurons, where n corresponds to the number of attributes of the considered traffic [Karsoliya 2012]. The hidden layer neurons, responsible for network learning, use the ReLU activation function [Agarap 2019]. This activation function returns the value received by the adder if it is greater than zero. If the values are negative, it returns zero. As it is a simple function, it has become a function widely used in different types of neural networks, as it is easier to train and usually achieve good results [Agarap 2019]. The output layer is composed of only one neuron that uses the sigmoid activation function.

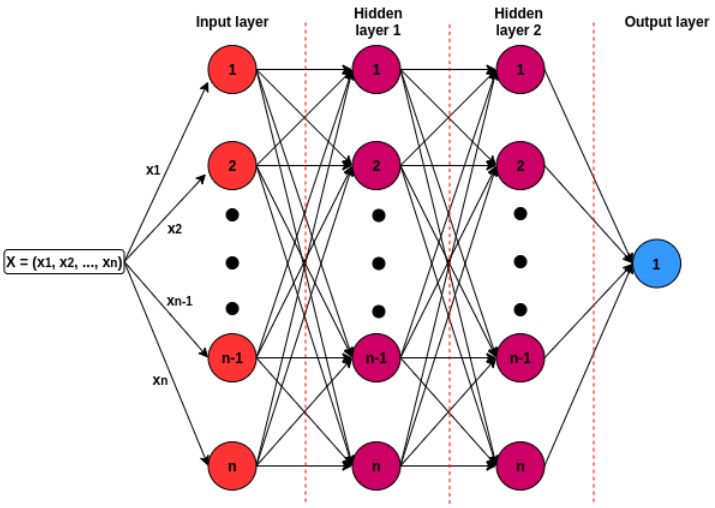

Figure 3. Defined architecture for DNN networks.

As mentioned before, the proposed method based on OVA uses K neural models, where each one will discriminate whether the input belongs or not to its respective class. Each neural model is trained to discriminate its respective class from all others. Input is routed to each of the MLP models, and the outputs of these neural models are then combined into a global decision that defines the final classification of the input. An instance is classified into the class whose corresponding classifier has the highest activation output. Mathematically, the decision function D can be defined according to Equation 1. Where

f_i is the output of activating an DNN trained with class i against other classes and x is a vector of input resources [Oong and Isa 2012].

$$D(x) = \underset{i \in \{1,\ldots,K\}}{arg\ max}\ f_i(x) \tag{1}$$

3.2. Training

The classification approach needs to carry out a training step to analyze and classify new data traffic. As can be seen in Figure 4, the data selected for training are formatted and processed to remove invalid records in a pre-processing step.

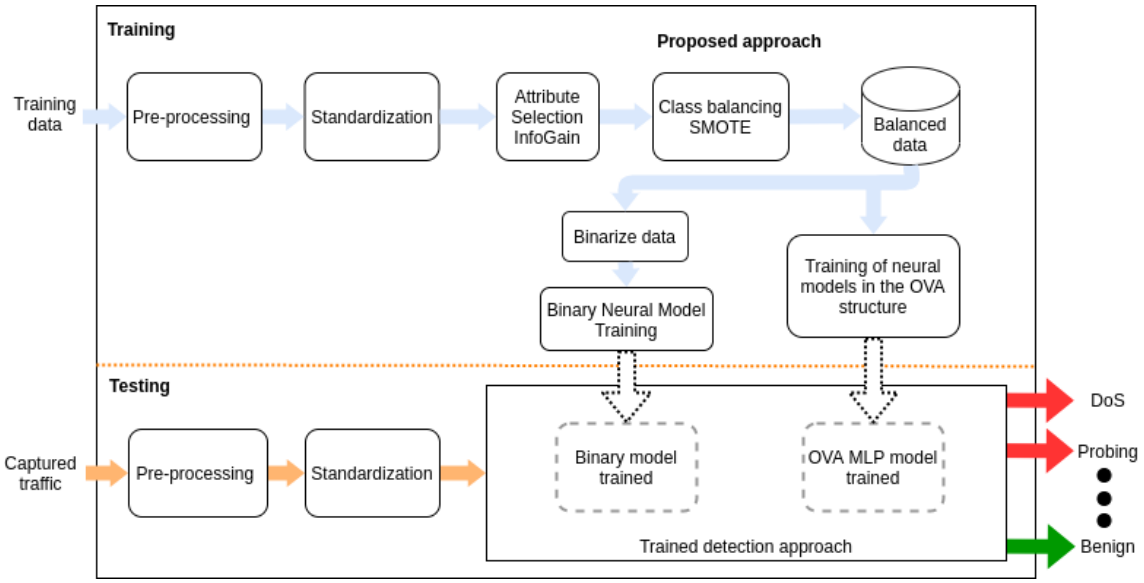

Figure 4. Complete overview of the proposed approach.

The next step refers to dataset standardization. This step helps to improve some machine learning-based classifiers that need their resources to be normally distributed. The method used for data normalization was the standard scaling, given by the Equation 2, with x the sample, u the mean, and s the standard deviation. The mean and standard deviation are obtained based on the statistics for each attribute in the dataset.

$$z = \frac{x - u}{s} \tag{2}$$

The attribute selection step aims to select the best traffic characteristics to be used by the classification method. Selecting only the best attributes can help improve classifier performance and also save resources. Attributes can be selected according to their relevance and quality to the ranking task. One way to measure the quality of an attribute is to assess its degree of association with the class by measuring Information Gain. The Information Gain of an A attribute of a given D dataset provides a measure of the expected entropy decrease when using the A attribute to partition the dataset [Quinlan 1986]. Entropy characterizes the impurity of the data in a set. It is a measure of the heterogeneity of the input data regarding its classification. The algorithm based on the Gain Ratio was used

to find the best characteristics of the traffic packets. The resource selection module based on this attribute selection extracts and sends to the detection module only the resources most relevant to the process.

Subsequently, the approach performs a data balancing process. This step is carried out to overcome the difficulty of working with unbalanced data, where some classes have a much higher amount of records than others. Therefore, the Synthetic Minority Oversampling Technique (*SMOTE*) [Bowyer et al. 2011] technique was used, which consists of synthesizing new records for classes that contain few records. In order not to generate a large amount of new data, in case of extremely unbalanced sets, the standard balancing approach was not used. In this work, we chose to propose a balancing strategy based on creating data so that all classes have at least 20% of the majority class data.

The next step consists of training the neural models under the OVA structure to generate the complete trained model. For this, the balanced data from the previous step are used. Furthermore, the balanced data is transformed into binary classes for training the binary neural model.

After the training process, the model will be ready to operate, receiving new data captured from the network to be classified. During actual execution, data is captured, preprocessed to obtain only selected attributes, standardized, and submitted to the trained approach, which then performs the detection and identification of the traffic category.

4. Evaluation

In this section, some aspects of the methodology used to evaluate the proposed approach are described. Initially, the classification metrics considered are presented. Then, the characteristics of the dataset and its division into sets for training and testing are presented. Finally, the results achieved through the experiments are presented and discussed.

4.1. Evaluation Metrics

For model evaluation, some evaluation metrics commonly used to evaluate classification applications will be used. Initially, the following measurements are obtained from the experiments: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). From the above measurements it is possible to calculate the following evaluation metrics [Liu and Lang 2019]:

- **Accuracy (ACC):** Proportion of correctly detected data over the total existing.
- **Precision (PRE):** Proportion of data correctly classified as intrusion to total data classified as intrusion.
- **Detection Rate (DR, Recall or Sensitivity):** Proportion of data classified as intrusion over total intrusion data.
- **Balanced accuracy:** Metric used in multiclass classification where the data is unbalanced, where the accuracy of the class is weighted by the number of instances.

4.2. BoT-IoT dataset

The BoT-IoT dataset was created by designing a realistic network environment in the Cyber Range Lab at UNSW Canberra Cyber Center. The environment incorporates normal traffic and attacks such as Denial of Service (DoS), Distributed Denial of Service

(DDoS), Reconnaissance, and Theft. The base has 46 attributes referring to traffic characteristics. This set is extremely recent and realistic for current traffic in the IoT context [Koroniotis et al. 2019]. In this work, a stratified subset of 1,089,696 instances of the Bot-IoT dataset was used. Table 2 presents the description and number of attacks present in the Bot-IoT dataset [Koroniotis et al. 2019]:

Table 2. Amount of data for each class in Bot-IoT dataset.

Class	Description	Amount of data
Benign	Benign traffic.	143
DDoS	There are numerous machines launching denial attacks against a victim at the same time.	574431
DoS	They aim to affect the victim's availability by flooding with a large volume of orders.	490482
Reconnaissance	Are malicious activities that gather information about victims by scanning remote systems.	24616
Data Theft	It is a group of attacks that seek to compromise the security of a machine to obtain confidential data.	24

This dataset has some control attributes that were removed to ensure a fair experiment, such as the *pkSeqID*, *attack*, *subcategory* attributes. In addition, some attributes that had redundancy in the set were removed, such as *flgs*, *proto*, *state*.

The Table 3 presents the attributes of the Bot-IoT database selected through the information gain rate. Only attributes that obtained a gain rate greater than 0.10 were considered, so in this case, 28 attributes were selected to be used in the classification process.

Table 3. Bot-IoT attributes selected through information gain rate.

	Attribute	Gain Ratio		Attribute	Gain Ratio
1	ltime	0,6931	15	daddr	0,3386
2	stime	0,6931	16	min	0,3263
3	dur	0,6852	17	TnBPSrcIP	0,2888
4	AR_P_Proto_P_DstIP	0,6818	18	sbytes	0,2562
5	AR_P_Proto_P_Dport	0,6808	19	bytes	0,2442
6	AR_P_Proto_P_SrcIP	0,6807	20	TnP_PDstIP	0,2020
7	AR_P_Proto_P_Sport	0,6755	21	TnP_Per_Dport	0,1979
8	rate	0,6638	22	TnP_PerProto	0,1837
9	srate	0,6490	23	N_IN_Conn_P_DstIP	0,1709
10	sum	0,5932	24	Pkts_P_State_P_Protocol_P_DestIP	0,1764
11	mean	0,5755	25	TnP_PSrcIP	0,1564
12	max	0,5668	26	Pkts_P_State_P_Protocol_P_SrcIP	0,1531
13	stddev	0,4892	27	spkts	0,1374
14	TnBPDstIP	0,3832	28	pkts	0,1337

To assess the feasibility and performance of the proposed approach, experiments were carried out with various machine learning techniques. To carry out these experiments, the data set is divided into 70% of the data for training classifiers and 30% of the data for testing the classifiers' performance. Five runs were performed for each experiment, and the results presented correspond to the average of the results obtained in the 5 runs. The classical techniques used in the comparison were: DNN, K-Nearest Neighbor (KNN), Extra Tree (ET), and Naïve Bayes (NB).

Table 4 presents the results obtained in the experiments with the Bot-IoT dataset. The results calculated from the metrics described above are displayed. The Naïve Bayes method presented greater difficulties in identifying normal traffic, being close to 90% of identification. These 10% not identified as normal traffic were therefore classified as attacks, which, as mentioned above, can become harmful to the network. The DNN approach and KNN showed approximately 95% of benign traffic identification. However, with DNN, it achieved only 93% precision. This indicates that almost 7% of the traffic identified as benign was actually an attack. That is, false negatives occurred, where the intrusive traffic would not be blocked, which could harm the functioning of the network.

Table 4. Results obtained by machine learning approaches and by the proposed approach with the Bot-IoT dataset.

	DNN		kNN		ET		NB		Our Work	
	PRE	**DR**	**PRE**	**DR**	**PRE**	**DR**	**PRE**	**DR**	**PRE**	**DR**
Benign	93.1	95.3	95.3	95.3	100.0	100.0	61.9	90.6	97.7	98.6
DDoS	99.9	100.0	99.9	99.9	99.9	100.0	100.0	99.4	100.0	100.0
DoS	100.0	99.9	99.9	99.9	100.0	100.0	99.5	99.9	100.0	100.0
Recon	99.9	99.9	99.9	100.0	100.0	100.0	95.2	99.6	99.9	99.9
Theft	100.0	100.0	100.0	71.4	100.0	71.4	100.0	71.4	100.0	100.0

It is observed that all techniques were able to achieve high attack detection rates such as DDoS, DoS, and Reconnaissance, being close to 100% of detection. However, KNN, ET, and NB approach detected the only 71% of data theft attacks. DNN had detection rates close to 100% for all classes of attacks. However, it had difficulties related to the identification of normal flow and false negatives. It is observed that the proposed approach presented detection rates (recall) greater than 98% in all classes, indicating a high capacity for classification correctness. In more detail, the approach achieved approximately 100.0% rate of DDoS, DoS, and Reconnaissance attacks. In addition, all data theft attacks were detected, indicating a recall of 100%. It can be observed that the proposed approach with two-stage detection architecture and the use of a multiclass method decomposed into several binary classifiers, together with class balancing and attribute selection technique, was able to provide greater stability in the detection rate metric. Furthermore, to achieving great recall rates, she was also able to obtain great precision. That is, the false-positive rate was low. As mentioned above, false positives are normal situations that have been misidentified as attacks. A large number of situations of this type can harm the functioning of the network as it will block legitimate traffic.

Next, in Table 5, the general average results among the classes are presented. Information on accuracy (ACC), balanced accuracy (BACC), precision (PRE), and detection time is provided. Also, Figure 5 presents a comparative graph of the main general metrics obtained by the proposed approach and machine learning techniques.

It is observed that in all cases, the overall accuracy and precision rates of each approach were excellent. However, these metrics may not reflect the actual detection rate for each class. Accuracy indicates the proportion of hits among all existing traffic, regardless of classes. In this case, DDoS and DoS attacks that achieved great detection rates and have many examples influence the overall metric more strongly. This can hide bad results from other classes with few instances and thus hardly influence the overall result, as is the case with Data Theft attacks. The general metric of balanced accuracy indicates the accuracy

Table 5. Overall results of experiments with the NSL-KDD data set.

	Accuracy (%)	Balanced accuracy (%)	Precision (%)	Detection Time (s)
DNN	99.99	99.06	99.99	0.52
kNN	99.99	93.35	99.99	245.99
ET	99.99	94.29	99.99	4.01
NB	99.66	92.23	99.67	0.63
Our work	99.99	99.72	99.99	4.83

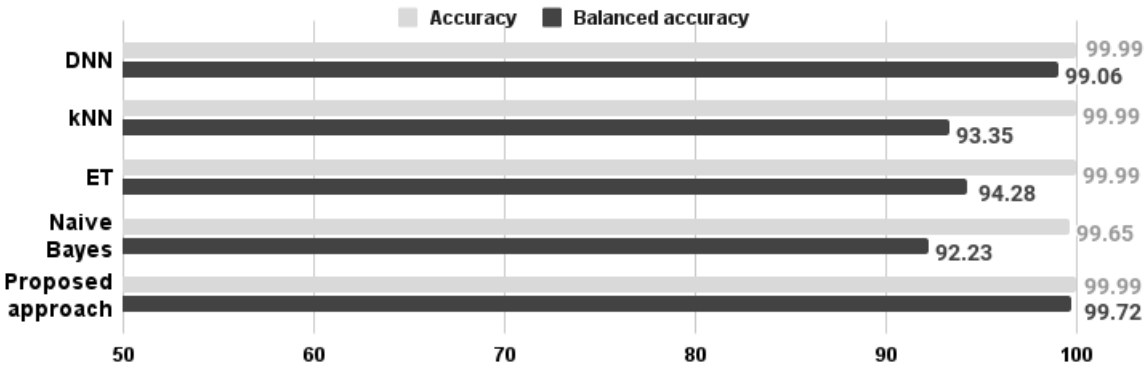

Figure 5. Results obtained by machine learning approaches and by the proposed approach with the Bot-IoT dataset.

considering a balance with the classes, so a negative or positive result of classes with few instances can reflect the general result. In this case, it appears that the proposed approach presented balanced accuracy of 99.7%, indicating that it maintained a pattern of correct answers in each class close to 99%. On the other hand, the other techniques presented difficulties in detecting some classes and presented less balanced accuracy. The KNN, ET, and NB techniques had difficulties mainly in detecting Data Theft attacks. DNN detected Data Theft attacks and achieved a balanced accuracy of 99%. However, it presented a lower identification and accuracy of benign traffic, which could harm the network.

The Detection time metric indicates the time required to classify the entire set of tests used in the experiment. The KNN approach had the longest time, about 245 seconds. This is mainly due to the numerous comparisons with the sets of examples that are performed during classification. NB obtained a time of 0.6 seconds. However, it presented inconsistent detection results. The ET achieved average detection results and took 4 seconds to perform the classification. DNN showed good detection results and was the approach with the shortest time, just 0.5 seconds. However, it presented inferior results of benign traffic. Finally, the proposed approach achieved the greatest balanced accuracy and harmony between the individual results. However, it proved to be more costly in relation to DNN and ET, requiring 4.8 seconds to perform the classification. However, it should be noted that the architecture of the proposed approach has the first level of binary detection, and only events detected as intrusive will be sent to the second level. That is, benign events will be released in this first analysis. As most of the real traffic is usually benign, the approach will approximate the detection time to DNN, as the first level of the approach consists of a simple DNN. This is not reflected in the time of

this experiment, as the data set used has many more intrusive examples. That is, the two steps of the approach are performed for most of the data, increasing the detection time.

Furthermore, in the future, the implementation of the second stage can be discussed in an environment with greater computational power, for example, cloud computing. As only data already detected as intrusive are submitted to the second stage, communication latency would not be so serious.

The results obtained by the proposed approach in the experiments with the Bot-IoT dataset are compared with the results obtained by the state-of-the-art approaches in Table 6. The work by Basset et al. [Abdel-Basset et al. 2020] kept detection rates above 97% for all attacks. The approach proposed by Soe et al. [Soe et al. 2020] obtained good results for three classes considered. However, it did not present the benign traffic detection rate and treated DDoS and DoS as a single category. Bhuvaneswari and Selvakumar [Bhuvaneswari and Selvakumar 2020] and Shafiq et al. [Shafiq et al. 2021] approachs performed well but had difficulty detecting data theft. On the other hand, the approach proposed by Popoola et al. [Popoola et al. 2021] achieved excellent attack detection rates but had some difficulty detecting benign traffic, which indicates an increase in the number of false positives. Overall, the proposed approach achieved the best detection rates across all categories of attacks, keeping all detection rates close to 99%.

Table 6. Detection rate obtained by state of the art works in experiments with the Bot-IoT dataset.

	Benign	**DDoS**	**DoS**	**Recon**	**Theft**
[Abdel-Basset et al. 2020]	97.2	99.8	99.9	99.9	97.4
[Bhuvaneswari and Selvakumar 2020]	90.5	99.7	99.8	99.9	77.2
[Soe et al. 2020]	-	99.4		99.3	99.3
[Popoola et al. 2021]	92.9	99.5	99.40	100.0	100.0
[Shafiq et al. 2021]	93.9	-	100.0	100.0	50
Our work	98.6	100.0	100.0	99.9	100.0

5. Conclusion

Intrusion detection is one of the main security points, aiming to identify attack attempts by malicious entities. In addition to detecting multiclass, it is important to carry out specialized countermeasures for each type of attack and assist in decision making by the person responsible for the network. However, existing multiclass detection approaches present some difficulties related to false positives and detection of less common attacks.

This work aims to propose an approach for the detection and identification of intrusions. The approach has a two-stage detection architecture, which allows a more robust method to discriminate attacks into categories. First, a binary MultiLayer Perceptron model is employed. Later, a multiclass approach based on One vs. All (OVA) and MLP neural models are used to discriminate intrusive traffic. In addition, the approach uses the SMOTE technique to create new records to balance training data. The results obtained through experiments with the Bot-IoT intrusion dataset demonstrate that the approach achieved promising results compared to machine learning methods and reduced false positives compared to state-of-the-art approaches. As future work, we highlight the improvement of the approach's deployment architecture and the need for more studies on automated and optimized countermeasure methods to mitigate specific types of intrusions.

References

Abdel-Basset, M., Chang, V., Hawash, H., Chakrabortty, R. K., and Ryan, M. (2020). Deep-ifs: Intrusion detection approach for iiot traffic in fog environment. *IEEE Transactions on Industrial Informatics*.

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., and Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11):e00938.

Agarap, A. F. (2019). Deep learning using rectified linear units (relu).

Almiani, M., AbuGhazleh, A., Al-Rahayfeh, A., Atiewi, S., and Razaque, A. (2020). Deep recurrent neural network for iot intrusion detection system. *Simulation Modelling Practice and Theory*, 101:102031. Modeling and Simulation of Fog Computing.

Bhuvaneswari, A. N. and Selvakumar, S. (2020). Anomaly detection framework for internet of things traffic using vector convolutional deep learning approach in fog environment. *Future Generation Computer Systems*, 113:255–265.

Bhuvaneswari Amma, N. G. and Subramanian, S. (2018). Vcdeepfl: Vector convolutional deep feature learning approach for identification of known and unknown denial of service attacks. In *TENCON 2018 - 2018 IEEE Region 10 Conference*, pages 0640–0645.

Bowyer, K. W., Chawla, N. V., Hall, L. O., and Kegelmeyer, W. P. (2011). SMOTE: synthetic minority over-sampling technique. *CoRR*, abs/1106.1813.

Chua, L. O. and Yang, L. (1988). Cellular neural networks: Theory. *IEEE Transactions on circuits and systems*, 35(10):1257–1272.

Conti, M., Dehghantanha, A., Franke, K., and Watson, S. (2018). Internet of things security and forensics: Challenges and opportunities.

Diro, A. A. and Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for internet of things. *Future Generation Computer Systems*, 82:761 – 768.

Du, R., Li, Y., Liang, X., and Tian, J. (2020). Support vector machine intrusion detection scheme based on cloud-fog collaboration. In *International Conference on Security and Privacy in New Computing Environments*, pages 321–334. Springer.

Frustaci, M., Pace, P., Aloi, G., and Fortino, G. (2017). Evaluating critical security issues of the iot world: Present and future challenges. *IEEE Internet of things journal*, 5(4):2483–2495.

Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., and Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *computers & security*, 28(1-2):18–28.

Haykin, S. et al. (2009). Neural networks and learning machines. *Upper Saddle River: Pearson Education*, 3.

Hughes, K., McLaughlin, K., and Sezer, S. (2020). Dynamic countermeasure knowledge for intrusion response systems. In *2020 31st Irish Signals and Systems Conference (ISSC)*, pages 1–6. IEEE.

Iorga, M., Feldman, L., Barton, R., and Martin, M. (2018). Fog computing conceptual model. special publication (nist sp)-500–325.

Karsoliya, S. (2012). Approximating number of hidden layer neurons in multiple hidden layer bpnn architecture. *International Journal of Engineering Trends and Technology*, 3(6):714–717.

Koroniotis, N., Moustafa, N., Sitnikova, E., and 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.

Liu, H. and Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. *Applied Sciences*, 9(20).

Lorena, A. C., De Carvalho, A. C., and Gama, J. M. (2008). A review on the combination of binary classifiers in multiclass problems. *Artificial Intelligence Review*, 30(1-4):19.

Oong, T. H. and Isa, N. A. M. (2012). One-against-all ensemble for multiclass pattern classification. *Applied Soft Computing*, 12(4):1303–1308.

Oppitz, M. and Tomsu, P. (2018). Internet of things. In *Inventing the Cloud Century*, pages 435–469. Springer.

Popoola, S. I., Adebisi, B., Hammoudeh, M., Gui, G., and Gacanin, H. (2021). Hybrid deep learning for botnet attack detection in the internet-of-things networks. *IEEE Internet of Things Journal*, 8(6):4944–4956.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, 1(1):81–106.

Shafiq, M., Tian, Z., Bashir, A. K., Du, X., and Guizani, M. (2021). Corrauc: A malicious bot-iot traffic detection method in iot network using machine-learning techniques. *IEEE Internet of Things Journal*, 8(5):3242–3254.

Soe, Y. N., Feng, Y., Santosa, P. I., Hartanto, R., and Sakurai, K. (2020). Towards a lightweight detection system for cyber attacks in the iot environment using corresponding features. *Electronics*, 9(1).

Vikram, N., Harish, K., Nihaal, M., Umesh, R., and Kumar, S. A. A. (2017). A low cost home automation system using wi-fi based wireless sensor network incorporating internet of things (iot). In *2017 IEEE 7th International Advance Computing Conference (IACC)*, pages 174–178. IEEE.