EVALUATION OF CYBERSECURITY DATA SET CHARACHTERISTICS FOR THEIR APPLICABILITY TO NEUTRAL NETWORKS ALGORITHMS DETECTING CYBERSECURITY ANOMALIES

[1] Y.Harshil, [2] E.Ramesh Reddy

yellaharshil@gmail.com, [2]rameshreddycse@gmail.com

2 Associate. Professor Department of CSE Narayana Engineering College Gudur,

Abstract: *Artificial intelligence algorithms have a leading role in the field of cybersecurity and attack detection, being able to present better results in some scenarios than classic intrusion detection systems such as Snort or Suricata. In this sense, this research focuses on the evaluation of characteristics for different wellestablished Machine Leaning algorithms commonly applied to IDS scenarios. To do this, a categoriza- tion for cybersecurity data sets that groups its records into several groups is first considered. Making use of this division, this work seeks to determine which neural network model (multilayer or recurrent), activation function, and learning algorithm yield higher accuracy values, depending on the group of data. Finally, the results are used to determine which group of data from a cybersecurity data set are more relevant and representative for the intrusion detection, and the most suitable configuration of Machine Learning algorithm to decrease the computational load of the system.*

.

Keywords: *Cybersecurity, data analytics, data sets, machine learning*

INTRODUCTION

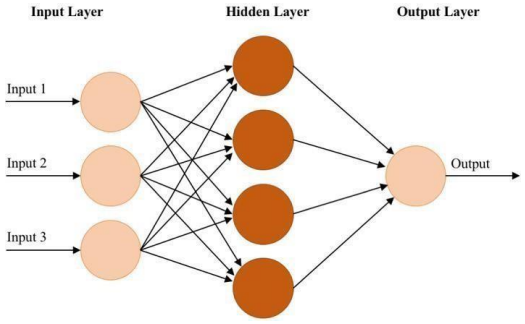
The increased complexity of new computer systems and the adaptability of new technological developments is leading to the progress in the application of new methods and tech- niques of artificial intelligence (AI) in the field of computer security. Specifically, AI has had a greater incidence in the detection of harmful software or anomalies and intrusions, generating new to modules to support more efficient and robust decisions [1]. This aid, among other things, That allows human interaction to focus on moabstract actions such as general monitoring of their systems or the analysis of errors, i.e., false pos- itives. In addition, AI techniques also help

people responsible for IT security to manage and analyze the vast quantity of data that new information can generate.

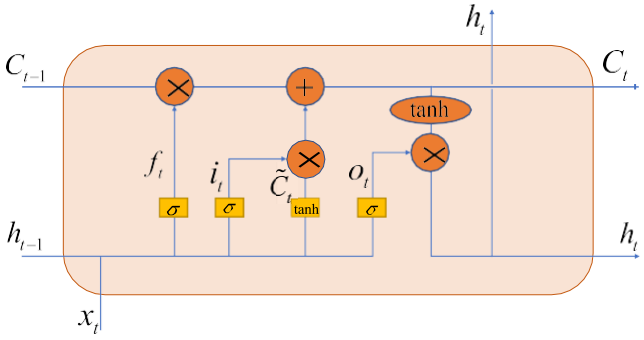
RELATEDWORK

.

Artificial neural networks are complex systems constructed by simple computational units called neurons, analogous to the behavior of neurons in biological brains. These neurons are interconnected through links that manage the activation state of adjacent neurons



A Network Model:-



Neutral & Neural Networks

In contrast to the feedforward neural networks, there are recurrent neural networks (RNNs) where the signals can travel in both directions, introducing loops in the network, which results in the output of a layer affecting this same layer, so it can give the network the memory property. Therefore, this type of neural network is generally used for the modeling of time series or tasks [4]. The use of RNNs is lower com- pared to feedforward networks, partly because the learning algorithms are much less effective (to date). However, they are presented as a very interesting alternative [5].

One of the most commonly used RNN architectures is the long short-term memory (LSTM) network [6], which minimizes the problem of gradient descent. Figure 2 presents the basic scheme of a processing unit of this type of neural network.

The key to understanding the functioning of these networks are the values $C_{t\ 1}$ and C_t, which represent the state of each cell. Thus, a cell can maintain its state in time (through the horizontal line that connects $C_{t\ 1}$ and C_t) regulating the flow of information between the input and the output through nonlinear doors

This work is licensed under a Creative Commons Attribution 4.0 International License

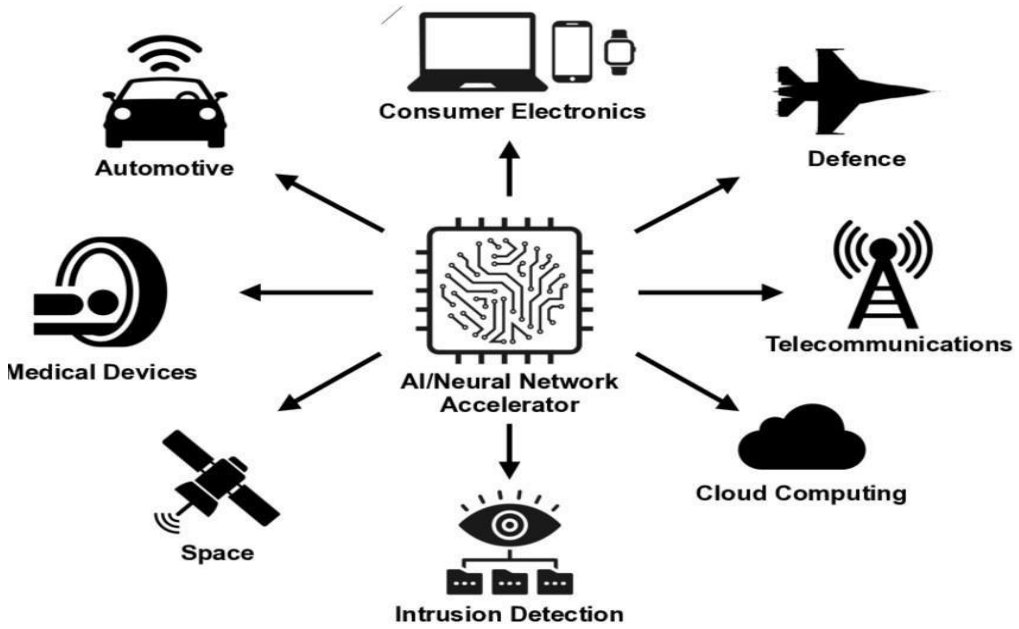
The uses of neural and neutral networks in cyber security

The application of artificial neural networks to the context of computer security is mainly focused on the detection of intrusions in a network since artificial neural networks are considered an efficient approach to pattern classification. The main problem with these algorithms consists of the high calculation requirements and the long training cycles they require, hindering their incorporation into commercial applications [16].

Even though different cybersecurity proposal are based on old public datasets, their results are not comparable due to different causes: different algorithms consider different fea- tures, implementation of pre-filtering operation and the use of different split between test and training data set [17]. For this, the current article makes an exhaustive comparation of dif- ferent groups of characteristics applying different algorithms of feed forward neural networks (FFNN) and RNN setting and testing different configurations proposed as mentioned in section VI

Artificial neural networks, as discussed above, have been used in multiple and diverse problems related to IDSs.

Another type of artificial neural network architecture, pop- ular for computer security environments, is recurrent net- works. In [25], an IDS architecture was presented where distributed-time deferred neural networks (DTDNN) were used, which provides a simple and efficient method for classi- fying data sets because of its high speed and fast convergence rates, with satisfactory results. Another type of recurrent neu- ral network architecture widely used for the development of IDS is the so-called long short-term memory (LSTM), which was introduced in [7], [4], [26], [27]. In [7], an accuracy of 97.54% was presented, which is equal to other neural network architectures but had a false positive rate of 9.98%, quite high, although below most others architectures of neural networks with which it was compared. Additionally, [26] presented an architecture that yields an overall accuracy of 93.72%, although, for recognition attacks, which were addressed in this work, the accuracy was very low (56.4%). The work performed in [4] achieved high accuracy in DoS attacks and normal connections but low performance in reconnaissance attacks, R2L and U2R. Finally, [27] presented results



The analysis of some existing data sets (UNB-ISCX-2012 [30], CTU-13 [31], MACCDC [32] or UGR'16 [33]) allows us to observe that they have different formats and feature, so that we can say that cybersecurity data sets are highly heterogeneous.

The methodology proposed in this case aims to sim- plify multidimensional data sets, choosing only the relevant characteristics for the specific scenario and thus making the learning algorithm lighter. Specifically, the novelty of this work is reducing this multidimensionality by groups of characteristics, instead of using an individual approach, as an alternative of those presented in the current state of the

art [34]–[37]. For this purpose, this research proposes three main feature groups: *basic connection characteristics, content characteristics, and traffic statistical characteristics*. Some of these characteristics will have more or less weight, depending on the type of attack being detected. For example, time-based traffic characteristics are especially useful for detecting high volumes of data in a small interval of time and, therefore, appropriate for possible denial-of-service (DoS) attacks. The following subsections describe each of them.

BASIC CONNECTION CHARACTERISTICS

This category includes the basic features that are usually found in a TCP header. They are intrinsic characteristics of a connection and can be useful for general-purpose network analysis, as well as being used for intrusion

The Data Set Understudy

For the problem addressed in this paper, the database used must contain information about different connections in a network together with a label that specifies whether the con- nection is an attack and its type or a normal connection. The algorithm used for detection will make use of supervised learning, and therefore, it is necessary that each type of data is labeled and classified.

In this case, the data set UNSW-NB15 [39], [40], which is widely used in cybersecurity [41]–[44] and considered as a benchmark data set [45], was chosen. The choice of this data set is motivated by several factors: the validity of the attacks the labeling of these, and the classification of the data, similar to that presented in the previous section.

The UNSW-NB15 data set is composed of 49 features, 47 of which are related to the attributes of the data; the last two features are related to the type of attack and the behavior in the data set (normal or attack).

Group	Feature
Basic characteristics	Srcip,sport,dstip,dsport,proto,state,dur,sbytes,dbytes, Stl,dttl,sloss,service,sload,dload,spkts,dpakts
Content characteristics	Swin,dwin,stcpb,dtcpb,smeansz,dmeansz,trans_depth Res_bdy_len
Time charachteristics based on time	Sjit,dijt
Traffic characteristics based on source code	ct_srv,ct_src,ct_src_dst_ltm
Traffic characteristics based on destination adress	ct_srv_dest, ct_dst_ltm,ct_dst_dport_ltm

Experimentation and discussion of results

For the implementation of neural networks, Python was used as the programming language, and the TensorFlow library, an open-source library created by Google Brain Team. This library offers all the necessary tools to build, train and test the effectiveness of artificial neural networks.

Throughout this section, the results obtained after the tests were carried out with the two neural network models, the activation functions, the different learning algorithms, and the different groups of characteristics are presented. In the different experiments tested, both the activation function of the neurons and the optimizer were modified.

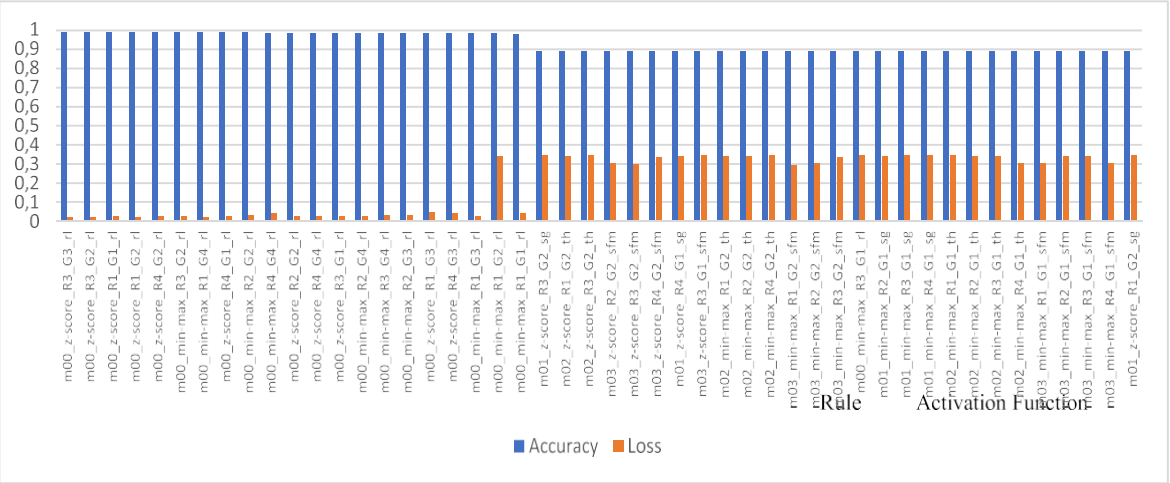
For experimentation and analysis, an *Adam* optimization algorithm was used. Once defined, the results related to the accuracy of the different activation functions were obtained. With these accuracy values,

the best activation function was selected, and several experiments were carried out with the optimizers. The variables that must be monitored to determine the performance of each network are the accuracy and the cost. To analyze the accuracy, we compared the test data for which connection labels were predicted, and the true val- ues of these labels, obtaining the total number of predicted labels by the algorithm and obtaining the percentage of accu- racy. Where appropriate, the cost focused on measuring the error between the test data for which connection labels were predicted, and the true value of the labels calculated the cross-entropy of the normalized exponential function. Once this error was obtained, it was averaged, and a value was obtained that was reduced in the next iteration of training. Finally, the weights of the neural network were initialized with random values.

A. ANALYSIS OF MULTILAYER NEURAL NETWORKS

The developed multilayer neural network consisted of three fully connected layers, an input layer, a hidden layer, and an output layer. The distribution of neurons of this network in each layer followed the set of rules defined in [48] and detailed in Table 3.

For the tests carried out in this research, the combinations of the activation function and the optimizers shown



in Table 3 were tested. For each configuration, both min-max and Z-score normalization were used. Additionally, each of these configurations was analyzed with each group of char- acteristics defined in Table 2, focusing on the best accuracy and determining the best configuration for each type of data. First, to determine the activation function, m00, m01, m02, and m03 tests were performed, using the optimizer Adam and both normalization functions, min-max and Z-score. The results of this experiment are shown in Figure 3 in a descen- dent form, from the best performances in terms of accuracy to the worst. These results indicate that the best results (higher accuracy and earlier convergence) were obtained by the use of a linear rectifier for all the groups of characteristics, obtaining values of approximately 98% of accuracy using the R_1 rule and the Z-score normalization function for each of the groups of characteristics. Similarly, it can be observed the activation function, the corresponding rule to determine the number of specific nodes for the hidden layers, as well as the most appropriate normalization function.

Then, in Figure 4, the experiments performed once the linear rectifier function is set, show that the maximum accu- racy was reached for each of the groups of characteris- tics of experiment m00, executed on each of the groups of characteristics and achieved an accuracy of 98.56%. The results of these tests are detailed in Table 5.For the data belonging to Group 2, the activation func- tion that achieved the best result was the linear rectifier, with an accuracy of 98.8%. Setting this as the activation function, the experiments related to the optimizers show that the highest accuracy was obtained with

the Adam optimizer (value indicated previously), followed by RMSProp, with an accuracy of 98.18%. The main disadvantage presented RMSProp was that the accuracy did not remain stable,

CONCLUSION

This work explored the application of neural networks to the detection of cybersecurity intrusions with two main objec- tives. First, the categorization of a data set (UNSW-NB15), dividing its characteristics into basic, content, traffic statis- tics and direction-based methods, to analyze which of these groups are the most relevant for the detection of anomalies, and to reduce training and reduce the loss of the models implemented. The second objective focused on determin- ing which neural network can offer a better performance according to the data available for its training.

The experiments performed, using the data set and the proposed categorization, allowed several conclusions to be drawn. The optimal results for each group of data were identified according to the type of neural net- work, the activation function, the optimization function, and the network architecture, as detailed in sections VI-a and VI-b. Additionally, the results show that when using only one group of data, an accurate prediction of the attack can be obtained, independent of the neural net- work topology. Thus, the configuration proposed as m00_z-score_R1_G1_rl obtained an accuracy similar to the configuration m00_z-score_R1_G4_rl, decreasing the load of the algorithm in terms of performance, but with a smaller number of characteristics, as detailed in VI-a.Regarding the comparison between the different neural network architectures analyzed, there was no substantial improvement when using recurrent networks instead of mul- tilayer networks, which was most likely due to the difficulty of training a recurring network.

Also, this article makes a comparison between different works that uses the same dataset and propose a similar idea in terms of characterization of features, selecting the most appropriate to get the best performance as possible in terms of accuracy. The results have exposed that our proposed model and characteristics have obtained the best accuracy with the FFNN and the group one of characteristics with 19 features. Finally, as future research drawn from this work, it is proposed, first, to extend the proposed methodology to a cybersecurity data set with more information, such as the one proposed by the Universidad de Granada in [26] (240M flows of data and real traffic).

Additionally, to improve the metric that estimates the performance of the algorithm, it is advisable to determine the types of predictions that are madeand not only obtain the percentage of the accuracy

REFERENCES

[1] B. Geluvaraj, P. M. Satwik, and T. A. Kumar, ‘‘The future of cybersecurity: Major role of artificial intelligence, machine learning, and deep learning in cyberspace,’’ in *Proc. Int. Conf. Comput. Netw. Commun. Technol.*, 2019, pp. 739–747.

[2] S. Dilek, H. Çakır, and M. Aydın, ‘‘Applications of artificial intel- ligence techniques to combating cyber crimes: A review,’’ 2015, *arXiv:1502.03552*. [Online]. Available: https://arxiv.org/abs/1502.03552

[3] A. Jović, K. Brkić, and N. Bogunović, ‘‘A review of feature selection meth- ods with applications,’’ in *Proc. 38th Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, 2015, pp. 1200–1205.

[4] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, ‘‘Long short term memory recurrent neural network classifier for intrusion detection,’’ in *Proc. Int. Conf. Platform Technol. Service (PlatCon)*, 2016, pp. 1–5.

[5] D. Berman, A. Buczak, J. Chavis, and C. Corbett, ‘‘A survey of deep learning methods for cyber security,’’ *Information*, vol. 10, no. 4, p. 122, Apr. 2019.

[6] S. Hochreiter and J. Schmidhuber, ‘‘Long short-term memory,’’ *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[7] T.-T.-H. Le, J. Kim, and H. Kim, ‘‘An effective intrusion detection classifier using long short–term memory with gradient descent optimization,’’ in *Proc. Int. Conf. Platform Technol. Service (PlatCon)*, Feb. 2017, pp. 1–6.

[8] T. Tieleman and G. Hinton, ‘‘Lecture 6.5-RMSPROP: Divide the gradient by a running average of its

recent magnitude,'' *COURSERA, Neural Netw. Mach. Learn.*, vol. 4, no. 2, pp. 26–31, 2012.

[9] D. P. Kingma and J. Ba, ''Adam: A method for stochastic optimiza- tion,'' Dec. 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv. org/abs/1412.6980

[10] J. Duchi, E. Hazan, and Y. Singer, ''Adaptive subgradient methods for online learning and stochastic optimization,'' *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Feb. 2011.

[11] M. D. Zeiler, ''ADADELTA: An adaptive learning rate method,'' 2012, *arXiv:1212.5701*. [Online]. Available: https://arxiv.org/abs/1212.5701

[12] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, ''On the importance of initialization and momentum in deep learning,'' in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1139–1147.

[13] D. Nikolov, I. Kordev, and S. Stefanova, ''Concept for network intrusion detection system based on recurrent neural network classifier,'' in *Proc. IEEE 27th Int. Sci. Conf. Electron. (ET)*, Sep. 2018, pp. 1–4.

[14] Y. Bengio, P. Simard, and P. Frasconi, ''Learning long-term dependencies with gradient descent is difficult,'' *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[15] V. Garcia-Font, C. Garrigues, and H. Rifà-Pous, ''Difficulties and chal- lenges of anomaly detection in smart cities: A laboratory analysis,'' *Sen- sors*, vol. 18, no. 10, p. 3198, Sep. 2018.

[16] N. Papernot, P. Mcdaniel, A. Swami, and R. Harang, ''Crafting adversarial input sequences for recurrent neural networks,'' in *Proc. IEEE Mil. Com- mun. Conf. (MILCOM)*, Nov. 2016, pp. 49–54.

[17] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, ''On the effectiveness of machine and deep learning for cyber security,'' in *Proc. 10th Int. Conf. Cyber Conflict (CyCon)*, May 2018, pp. 371–390.

[18] B. Subba, S. Biswas, and S. Karmakar, ''A neural network based system for intrusion detection and attack classification,'' in *Proc. 32nd Nat. Conf. Commun. (NCC)*, Mar. 2016, pp. 1–6.

[19] M. Moradi and M. Zulkernine, ''A neural network based system for intru- sion detection and classification of attacks,'' in *Proc. IEEE Int. Conf. Adv. Intell. Syst.-Theory Appl.*, 2004, pp. 15–18.

[20] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, ''A detailed analysis of the KDD CUP 99 data set,'' in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1–6.

[21] R. K. Vigneswaran, R. Vinayakumar, K. Soman, and P. Poornachandran, ''Evaluating shallow and deep neural networks for network intrusion detec- tion systems in cyber security,'' in *Proc. 9th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2018, pp. 1–6.

[22] R. Vinayakumar, H. B. B. Ganesh, P. Poornachandran, M. A. Kumar, and K. P. Soman, ''DeepNet: Deep neural network for cyber security use cases,'' 2018, *arXiv:1812.03519*. [Online]. Available: https://arxiv. org/abs/1812.03519

[23] O. Al-Jarrah and A. Arafat, ''Network intrusion detection system using neural network classification of attack behavior,'' *J. Adv. Inf. Technol.*, vol., 6, no. 1, pp. 1–8, 2015.

[24] C. Obimbo, K. Ali, and K. Mohamed, ''Using IDS to prevent XSS attacks,'' in *Proc. Int. Conf. Secur. Manage. (SAM)*, 2017, pp. 233–239.

[25] J. Deny and M. Sundhararajan, ''Neural networks and machine learning techniques for intrusion detection system,'' *Int. J. Digit. Commun. Netw.*, vol. 2, no. 1, pp. 5–8, 2015.

[26] R. C. Staudemeyer, ''Applying long short-term memory recurrent neural networks to intrusion detection,'' *South Afr. Comput. J.*, vol. 56, no. 1, Sep. 2015.

[27] C. Yin, Y. Zhu, J. Fei, and X. He, ''A deep learning approach for intru- sion detection using recurrent neural networks,'' *IEEE Access*, vol. 5, pp. 21954–21961, 2017.