A Framework for implementing an ML or DL model to improve Intrusion Detection Systems (IDS) in the NTMA context, with an example on the dataset (CSE-CIC-IDS2018).

Hakim Azeroual[1], *Imane Daha Belghiti*[2] *and Naoual Berbiche*[3]
LASTIMI, EST Sale, Mohammed V University in Rabat, Morocco
[1]hakimazeroual@research.emi.ac.ma
[2]imanedaha@gmail.com
[3]nberbiche@hotmail.com

Abstract. The objective of this work is to present a framework to be followed to model, test, validate and implement a DL model for anomaly, abuse, malware or botnet detection, with the aim of implementing or improving an Intrusion Detection System (IDS) within the NTMA framework, by means of new machine learning and deep learning techniques, which addresses reliability and processing speed considerations.

The said process will be used to perform studies on ML and DL models used for cybersecurity in isolation and in combination to extract conclusions, which can help in the improvement of intrusion detection systems using massive data collection techniques used in Big-Data.

The example discussed in this work implemented part of our framework by applying the CNN algorithm on the CSE-CIC-IDS2018 dataset. The results are encouraging for the use of ML in IDS, with an efficiency that exceeds 92% after 30 iterations. Thus, this model remains to be improved and tested on real networks.

Keywords— IDS – NIDS – NTMA - Deep Learning-Machine Learning - KDD Cup '99 - NSL-KDD - UNSW NB15 – Big Data – CNN.

1 Introduction

Cyber security is the set of policies, techniques, technologies, and processes that work together to protect the confidentiality, integrity, and availability of computing resources, networks, software programs, and data from attack. Cyber defense mechanisms exist at the application, network, host and data levels. There is an array of tools, such as firewalls, anti-virus software, intrusion detection systems (IDS) and intrusion protection systems (IPS), that work in silos to prevent attacks and detect security breaches. However, many adversaries still have an advantage because all they need to do is find a vulnerability in the systems to be protected.

Throughout the lifecycle of an attack, there are indicators of compromise; there may even be significant signs of an impending attack. The challenge is to find these indicators, which may be distributed throughout the environment. There are massive amounts of data from applications, servers, smart devices and other cyber resources generated by machine-to-machine and human-to-machine interactions. Cyber defense systems generate big data, such as the security information event management system (SIEM), which often overwhelms the security analyst with event alerts.

Using data science in cyber security helps correlate events, identify patterns and detect anomalous behavior to improve the security posture of any defense program. We are beginning to see the emergence of cyber defense systems that leverage data analytics. For example, network intrusion detection systems (NIDS), which inspect packet transmissions, are moving from signature-based systems, which detect known attacks, to anomaly-based systems, which detect deviations from a "normal" behavior profile.

Although Deep-Learning is a subdivision of machine learning, it is a newer and more complex method of learning than the norm. As such, the focus is on a thorough description of Deep-Learning methods and references to foundational work for each deep learning method are provided.

This paper is organized as follows: section 2 presents related work, section 3 explains the different DL methods used in cyber security, section 4 details the cyber security datasets for DL, section 5 describes some classification metrics used, and section 6 presents a Framework to follow in order to deploy a DL method for a specific purpose.

* Corresponding author : hazeroual11@gmail.com

2 Related work

In this section, a set of papers have been selected so as to list works that have cited The Big Data approach for collecting traffic data in networks, in addition to the use of DL in analysis and intrusion detection.

This paper [1] proposes a distributed framework based on the Big Data approach, in which storage and computational resources can be scaled to collect and process traffic in a large-scale network in a reasonable time.

In [2], the author gives a survey on the effectiveness of deep learning in analyzing and discovering knowledge in big data systems to recognize hidden and complex patterns. Through these successes, researchers in the field of networks apply deep learning models to the application of network traffic monitoring and analysis (NTMA).

Again in [3] the paper provides a brief tutorial-like description of each DL method is provided, including deep autoencoders, restricted Boltzmann machines, recurrent neural networks, generative adversarial networks and several others. In addition, [4] This paper presents a complete example of the steps involved in testing a new deep learning technique for intrusion detection. It also proposes our new deep learning classification model, evaluated using the KDD Cup '99 and NSL-KDD benchmark datasets. the technique demonstrates improvements over existing approaches and a strong potential for use in modern NIDS.

Fig. 1. Caption of the Figure 1. Below the figure.

3 Overview of the different DL methods used in cyber security

Machine learning techniques, including DL algorithms, are among the techniques for processing network traffic data. This is probably because modern communication systems and networks, have distinctive characteristics that are suitable for machine learning algorithms. These characteristics include big data generation, complexity, multimodal data, large scale, increasing number of protocols in these networks.

This paper [2] gives sub-areas for DL usage to meet the different needs of NTMA.

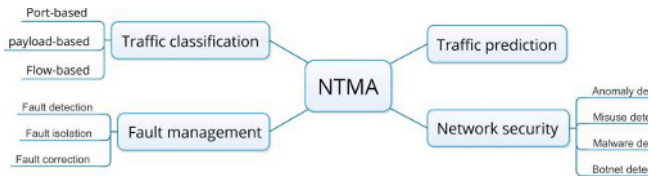

Fig 1 : Subdomains of DL use in the NTMA

Among the cyber defense tools, we can mainly mention firewalls, antivirus software and intrusion detection systems.

In our study on Network Traffic Monitoring and Analysis (NTMA), we will focus on network security and more specifically on intrusion detection systems (IDS).

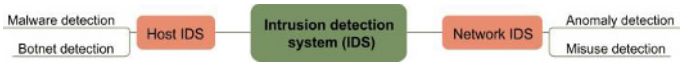

Fig 2: Intrusion detection systems (IDS)

To meet the needs of intrusion detection systems (IDS), the article [3],[5] present the test of several models in order to determine their contribution in the detection of intrusions on a data set whose types of attack are known in advance.

The following table 1 summarizes the results and conclusions of some models tested in isolation or combined.

Category	DL model	Key contribution
Misuse detection in modern networks	AE+ sparse AE + MAPE-K	Introduces a scalable, self-adaptive and autonomous method for misuse detection for modern large-scale modern networks by leveraging DL.
Anomaly detection system	CNN, AE and RNN	Designs and implements anomaly detection models based on different DL algorithms. Also, evaluates these models through standard classification metrics.
Anomaly detection	FCNs, VAE, and Seq2Seq	Examines multiple DL models for anomaly detection, including FCN, VAE, and LSTM.
Anomaly detection cloud datacenter	CNN+GWO	Proposes a robust hybrid method based on CNN and GWO for network anomaly detection in cloud environments, especially for streaming data.
Malware detection	CNN, RNN and LSTM	Introduces a multi-level DL system by using different DL models for malware detection.
Malware detection	SAEs	Introduces a two-phase framework for malware detection based on SAEs model.
Malware detection for IoBT	OpCode+ deep Eigenspace learning	Proposes the first work based on OpCode deep learning technique for IoT and IoBT malware detection.
Botnet detection	SDAs+ feed-forward supervised DNN	Discusses the application of DL for botnet detection and proposes a DL-based approach for botnet detection which utilizes TCP/UDP/IP packet flows as inputs.
Botnet detection	CNN+LSTM	Provides a botnet detection method, in which both network flow information and DL are used. Moreover, it uses graph structure for feature extraction purposes.
Botnet detection for IoT networks	Autoencoders	The first work that uses autoencoders in IoT networks for detecting botnet attacks. Also, unlike previous papers that use emulated or simulated data, this paper deploys real IoT traffic data for evaluation its method.
Anomaly and malware detection	Autoencoders	Provides an unsupervised feature learning method based on AE for cybersecurity purposes, e.g., anomaly and malware detection.
Anomaly detection in IEEE 802.11 network	SAE	One of the few articles that consider anomaly detection in the IEEE 802.11 network through DL.
Attack detection in MEC	DBNs	Provides a feature learning model based on deep belief network to detect attacks in MEC.
Anomaly detection in VANETs	GANs	Proposes a collaborative methods by leveraging deep generative models and distributed SDN to detect anomalies in VANETs.
Botnet detection	RNN	Analyzes the performance of RNN for botnet detection purposes through the behavioral analysis of network traffic.
Attacks detection in MCC	GRBM	Leverages the GRBM network to develop an attack detection method for mobile cloud environments.
Virtual MAC spoofing detection	CNN	Proposes a DL based detection system for MAC spooling attacks detection in virtualized environments.

4 Cyber security datasets for the DL

The datasets cited in the papers [1], [2] are used to evaluate the model and determine whether it can be used to detect attacks accurately or not. The quality of the dataset ultimately affects the outcome of any network intrusion detection system (NIDS). We consider among the best-known sets three datasets named KDD Cup'99, NSL-KDD, UNSW-NB15 [6], and WSN-DS. A detailed description of their characteristics is given below.

4.1 KDD Cup'99 Data set

KDD'99 data set was created by DARPA in 1999 by using recorded network traffic from 1998 dataset. It is being pre-processed into 41 features per network connection. Features in KDD'99 data set are categorized into four groups, Basic Features, Content Features, Time based traffic features, and Host based traffic features. KDD'99 consists of 4,898,430 records that is larger than other data sets. There are four main categories of attacks, these are DoS, R2L (unauthorized access from a remote machine), U2R (Unauthorized access to Root) and Probe. Many data mining techniques has been applied to the KDD'99 data set to detect intrusions in network traffic.

KDD Cup'99 is mostly used data set to build intrusion detection system (IDS). KDD data set have two critical issues concluded by the statistical analysis, that is profoundly affect the performance of the system. Most significant issue in KDD data set is that it has large number of replicated records. It is found that about 78% and 75% records are duplicate in train and test data set respectively. Huge number of replicated records may lead learning algorithms to be partial instead of numerous records. Thus, algorithm will stop learning infrequent records. These records may be harmful to network like U2R, R2L etc.

4.2 UNSW-NB15 Data set

The UNSW-NB15 dataset [7]is new and was published in 2015. It includes moderns attack (nine attack types compared to 14 attack types in KDD'99 dataset). It has 49 features and a variety of normal and attacked activities including with class labels of total 25,40,044 records. There are 2,21,876 normal records and 3,21,283 attacked records in the total number of records. Features of UNSW-NB15 data set is categorized into six groups namely Basic Features, Flow Features, Time Features, Content Features, Additional Generated Features, and Labelled Features. Further, UNSW-NB15 dataset has nine type of attacks category known as the Analysis, Fuzzers, Backdoors, DoS Exploits, Reconnaissance, Generic, Shellcode, and Worms.

4.3 CSE-CIC-IDS2018: A Dataset for Intrusion Detection Systems

This dataset was originally created by the University of New Brunswick to analyze DDoS data [8]. This dataset is entirely from 2018 and will not be updated in the future, however, new versions of the dataset will be available at the link above. The dataset itself was based on the university's server logs, which found various DoS attacks throughout the publicly available period. When using this dataset in machine learning, note that the Label column is arguably the most important part of the data, as it determines whether sent packets are malicious or not. See the Column Structures heading below for more information on this and other columns.

In total, there are 80 columns in this dataset, each corresponding to an entry in the IDS logging system that the University of New Brunswick has in place. Since their system classifies both forward and backward traffic, there are columns for both. The most important columns of this data set are listed below.

- Dst Port (Destination port) ;
- Protocol;
- Flow Duration;
- Tot Fwd Pkts (Total forward packets);
- Tot Bwd Pkts (Total backward packets);
- Label (Label).

5 Describes some classification metrics used.

To increase the performance of the model [9]; accuracy, recall, the precision rate should be calculated. We choose accuracy, recall, precision, and, F1 Measure for evaluation.

If we could create a confusion matrix [4], then it will be easy to calculate all the performance measures. Accuracy (1) is the percentage of true detection over total instances. Recall is how often does it predicted correct. Recall (2) is also known as True Positive Rate (TPR) or Sensitivity. Precision (3) tells that when it is predicted correct, how often is it actually correct. F1 measure (4) is a weighted average of the recall and precision. Mathematical representation of all measures can be extracted from the confusion matrix. In Figure 2, Actual No means actually normal records, Actual Yes means attacked records in actual, Predicted No means records that are predicted as normal and, Predicted Yes means records that are predicted as an attack. Confusion matrix is a table that is related to represent the performance of a classification model on a set of test data for which the true values are identified.

Total instances	**Predicted NO**	**Predicted YES**	
Actual NO	TN (True Negative)	FP (False Positive}	
Actual YES	FN(False Negative)	TP (True Positive)	**Recall**
		Precision	**Accuracy**

Fig 2 : The confusion matrix

$$\text{Accu\$\%c\&} \left(\frac{(T\# \,\%\, T\&)}{(T()*{+}ins)*n/0s)}\right) \qquad (1)$$

$$)*c\%{+}{+} \left(\frac{(T\#)}{(T()*{+}1/)2{+}340s)}\right) \qquad (2)$$

$$P\$*cisi/0\ (\frac{(T\#)}{(T()*+\#506i/)06478)} \quad (3)$$

$$12\ 3*\%su\$*\ (4*\frac{(\#50/isi(n*:0/*++)}{(\#50/isi(n\%:0/*++)} \quad (4)$$

6 Proposed framework for deploying a DL method for NTMA

In this section, we present a Framework to follow to model, test, validate and implement a DL model to detect anomalies[10], abuse, malware or botnets, with the objective of implementing or improving an Intrusion Detection System (IDS) in the NTMA framework [11].

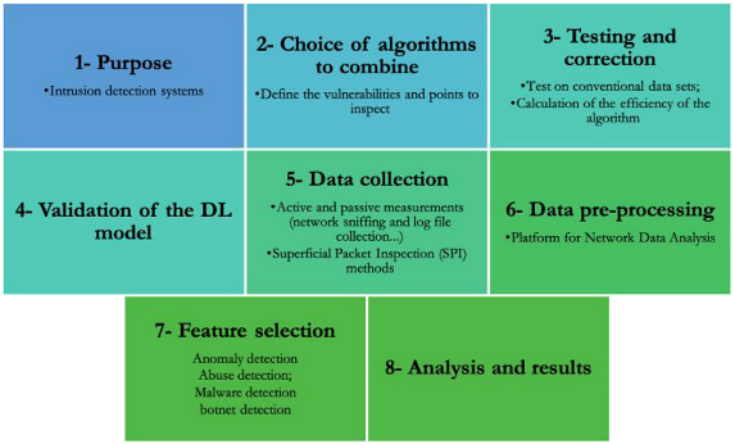

Fig 3 : Framework for validation and implementation of a DL model in NTMA.

7 Network Intrusion Detection Using Machine Learning/Deep Learning CNN on the dataset (CSE-CIC-IDS2018)

7.1 Data description

The dataset CSE-CIC-IDS2018[8] includes seven different attack scenarios: Brute-force, Heartbleed, Botnet, DoS, DDoS, Web attacks, and infiltration of the network from inside. The attacking infrastructure includes 50 machines and the victim organization has 5 departments and includes 420 machines and 30 servers. The dataset includes the captures network traffic and system logs of each machine, along with 80 features extracted from the captured traffic using CICFlowMeter-V3.

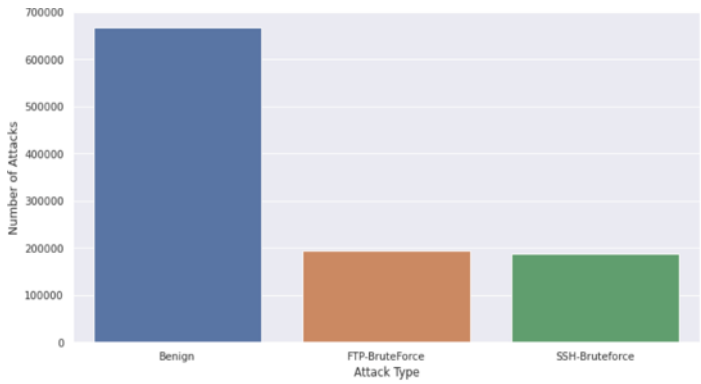

Fig 4 : Distribution of attack types.

For applying a convolutional neural network on our data, we will have to follow following steps:

- Separate the data of each of the labels
- Create a numerical matrix representation of labels
- Apply resampling on data so that can make the distribution equal for all labels
- Create X (predictor) and Y (target) variables
- Split the data into train and test sets
- Make data multi-dimensional for CNN

7.2 Data description

One of the most popular deep neural networks is the Convolutional Neural Network (CNN). It takes this name from mathematical linear operation between matrixes called convolution. CNN [12] have multiple layers; including convolutional layer, non-linearity layer, pooling layer and fully-connected layer. The convolutional and fully-connected layers have parameters but pooling and non-linearity layers don't have parameters. The CNN has an excellent performance in machine learning problems. Specially the applications that deal with image data, such as largest image classification data set (Image Net), computer vision, and in natural language processing (NLP) and the results achieved were very amazing.

7.3 Test and result

7.3.1 Data Splicing

This stage involves the data split into train & test sets. The training data will be used for training our model, and the testing data will be used to check the performance of model on unseen dataset. We're using a split of 80-20, i.e., 80% data to be used for training & 20% to be used for testing purpose.

7.3.2 Check the model performance on test data

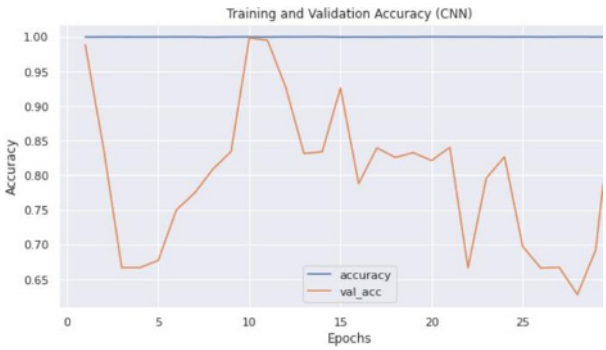

Fig 5 : Visualization training and the value of accuracy.

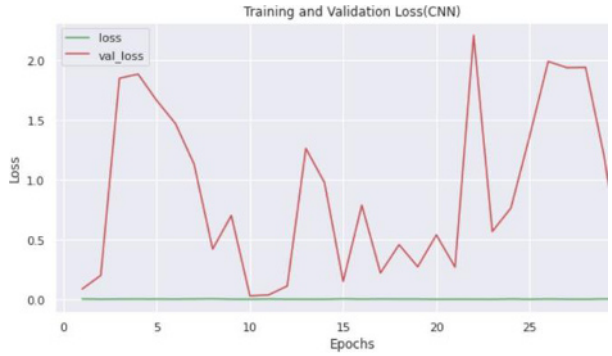

Fig 6 : Visualization of training and Validation Loss.

After training our deep CNN model on training data and validating it on validation data, it can be interpreted that:

- Model was trained on 50 epochs and then on 30 epochs
- CNN performed exceptionally well on training data and the accuracy was 99%
- Model accuracy was down to 83.55% on validation data after 50 iterations, and gave a good accuracy of 92% after 30 iterations. Thus, it can be interpreted that optimal number of iterations on which this model can perform are 30.

8 Conclusions

The framework presented will trace the process that will be followed during our research work to model, test, validate and implement an ML or DL model to improve an intrusion detection system (IDS) in surveillance and monitoring. network traffic analysis (NTMA) and that will meet the considerations of reliability and processing speed required by the massive data collection techniques used in Big-Data.
The example used in this work allowed us to implement the first 3 steps of our Framework, by applying the CNN algorithm on the CSE-CIC-IDS2018 dataset. The results are encouraging for the use of ML in IDS, with an efficiency that exceeds 92% after 30 iterations. Thus, this ML model remains to be compared, improved and tested on real networks in the context of our project.

References

1. L. U. Laboshin, A. A. Lukashin, and V. S. Zaborovsky, in *Procedia Computer Science* (Elsevier B.V., 2017), pp. 536–542
2. M. Abbasi, A. Shahraki, and A. Taherkordi, Computer Communications **170**, 19 (2021)
3. D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, Information (Switzerland) **10**, (2019)
4. Q. Niyaz, W. Sun, A. Y. Javaid, and M. Alam, in *EAI International Conference on Bio-Inspired Information and Communications Technologies (BICT)* (2015)
5. B. Mahbooba, R. Sahal, W. Alosaimi, and M. Serrano, Complexity **2021**, (2021)
6. S. Choudhary and N. Kesswani, in *Procedia Computer Science* (Elsevier B.V., 2020), pp. 1561–1573
7. N. Moustafa and J. Slay, *UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 Network Data Set)* (n.d.)
8. J. L. Leevy and T. M. Khoshgoftaar, Journal of Big Data **7**, (2020)
9. N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, IEEE Transactions on Emerging Topics in Computational Intelligence **2**, 41 (2018)
10. N. Thapa, Z. Liu, A. Shaver, A. Esterline, B. Gokaraju, and K. Roy, Electronics (Switzerland) **10**, (2021)
11. I. Kotenko, I. Saenko, and A. Branitskiy, *Applying Big Data Processing and Machine Learning Methods for Mobile Internet of Things Security Monitoring* (n.d.)
12. O. Bayat, S. Aljawarneh, H. F. Carlak, International Association of Researchers, Institute of Electrical and Electronics Engineers, and Akdeniz Üniversitesi, *Proceedings of 2017 International Conference on Engineering & Technology (ICET'2017) : Akdeniz University, Antalya, Turkey, 21-23 August, 2017* (n.d.)

[1]