Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 7, July 2021: 3314-3321

Research Article

Extreme Machine Learning Algorithm and Parallel Processing Approach for Intelligent Intrusion Detection System

B Santhosh Kumar[1], M Raghavendra Reddy[2]

Abstract

An IDS using machine learning is under development that aims to provide network- and host-level intrusion detection in a timely and autonomous manner. However, because harmful assaults are always evolving and occuring in high numbers, various issues develop, necessitating a scalable solution. Various malware datasets are publicly available for further investigation by the cyber security community. However, no previous study has examined the performance of several machine learning algorithms using publicly available datasets in depth. Because malware is dynamic, with constantly changing attack tactics, publically available malware datasets must be updated and benchmarked on a regular basis. Deep neural network (DNN), a sort of deep learning model and EML is investigated in this research in order to construct a flexible and effective IDS for detecting and classifying unanticipated and unanticipated cyber-attacks. Because of the constant change in network behaviour and the quick evolution of attacks, it is required to analyse numerous datasets that have been created throughout time using static and dynamic methodologies. This sort of research aids in the identification of the optimal algorithm for identifying future cyber-attacks. Several freely accessible malware datasets demonstrate the evaluations of trials for DNNs and other conventional machine learning classifiers. Using KDDCup 99 dataset, the best network parameters and network topologies for DNNs are determined. With learning rate varied between 0.01 and 0.5, all DNN experiments are done for at least 1,000 epochs. Other datasets, such as NSL-KDD, UNSW-NB15, Kyoto, WSN-DS, and CICIDS 2017, are benchmarked using the DNN model which fared well on KDDCup 99. Using several hidden layers, our DNN model learns the complex and high-dimensional feature representation of

[1] Associate Professor, G. Pulla Reddy Engineering College(Autonomous) : Kurnool, AP, India
[2] Assistant Professor, G Pulla Reddy Engineering College (Autonomous) : Kurnool, AP, India

the IDS data. It has been confirmed through extensive experimentation that DNNs perform better than classifiers made of standard machine learning algorithms. Additionally, we build a framework named Scale-Hybrid-IDS-AlertNet (SHIA) that is simple to scale and easy to use for real-time network traffic monitoring and for on-the-fly host-level cyber attack alerting. Extension**:** In this project as extension work I added Extreme Machine Learning algorithm and parallel processing to get better accuracy and less execution time.

Keywords: *Cyber Security, Intrusion Detection, Malware, Big data, Machine Learning, Deep Learning, Deep Neural Networks, cyber attacks, Cybercrime, Machine Learning Algorithm and Parallel Processing*

Introduction

There is a possibility that at least $70 million was lost due to information and communications technology (ICT) systems and networks failing because of numerous attacks by both internal and foreign attackers. Due to this evolution in cyber-attacks, along with hardware, software, and network topologies like the growth of the Internet of Things (IoT), these cyberattacks are rather advanced. Attackers can wreak havoc by carrying out malicious cyber-attacks, which necessitate the use of an effective, dependable, and adaptable intrusion detection system (IDS). An intrusion detection system (IDS) is a tool that automatically detects and classifies intrusions, assaults, or security policy breaches, without human intervention, at network and host levels. Network-based intrusion detection systems (NIDS) and host-based intrusion detection systems (HIDS) are based on invasive activities, and so may be classed as such. A network intrusion detection system is termed an NIDS. When mirroring by networking devices, such as switches, routers, and network taps, are used, network behaviours are recorded and filtered in order to reveal hidden assaults and prospective dangers in network traffic. A HIDS is a type of IDS system which uses system activity, which might be represented by various log files, to spot assaults. Local sensors capture the log files. In contrast to NIDS, which examines the contents of each packet in network traffic flows, HIDS bases its monitoring on information recorded in various log files, including those associated with sensors, system, software, and file systems. The usage of both NIDS and HIDS is common in many businesses. This methodology uses abuse detection, anomaly detection, and stateful protocol analysis to analyse network traffic flows. The system is configured to detect attacks and may employ signatures and filters to identify when an attack is taking place. Signature databases are updated regularly by input from human beings. In the case of unknown assaults, this strategy is absolutely ineffectual. Malicious actions are detected by utilising heuristic methods. Most settings create a high proportion of false positives using anomaly detection [5]. An organisation is usually equipped with both the abuse and anomaly detection techniques in its commercial solution systems to fight this problem. Because stateful protocol analysis deals with the network, application, and transport layers, it is far more powerful in contrast to the other detection approaches, which just focus on the network and application layers. It looks for deviations in the established vendor settings for supported protocols and apps. Though machine learning approaches have begun to be used in recent years to help detection systems recognise such types of attacks, there is a dearth of publicly available datasets to help with machine learning algorithm evaluation.

Some of the common issues that exist in existing machine learning solutions include: one, the models tend to have a high false positive rate; two, the models' performance is only reported for a single dataset; three, the machine learning models studied up to now have never dealt with today's complex network traffic; and finally, the solutions being proposed. These are manually and automatically executed, unique and continuously evolving in obscurity, leading in data breaches that go unnoticed. More specifically, in the case of the Yahoo data breach, which had lost $350 million, today's fast rising high-speed network size, speed, and dynamics are what caused the breach. Classical machine learning classifiers and deep neural networks (DNNs) applied to NIDS and HIDS are the difficulties that motivate this study. This research makes the following assumptions ;

Here to implement this paper I am using KDD and NSL dataset combination and I am using SVM, Random Forest and DNN algorithm with input hidden layer as 8. DNN algorithm keep filtering training algorithm with hidden layer to form most accurate model to predict testing class. DNN is a famous algorithm which has high predicting ratio in all fields such as image processing, data classification etc.

The following list comprises stages that compromise a target: an attacker's viewpoint

In general, unauthorized users known as attackers are responsible for network invasions. An attacker can render a service or machine completely inoperable remotely over the Internet or the attacker can attempt to access a machine remotely. With reliable intrusion detection, knowing how to properly hack a system is required. The classification of an assault generally consists of five distinct phases. Reconnaissance, exploitation, reinforcement, consolidation, and plundering all serve the goal of military strategy. A system can be hacked when it reaches the fourth or fifth phase. Therefore, it is extremely difficult to tell the difference between a natural occurrence and an attack. During Reconnaissance, an attacker tries to learn everything she can about the host and service reachability, as well as the OS and application versions.

Methodology

EXTREME MACHINE LEARNING (EML)

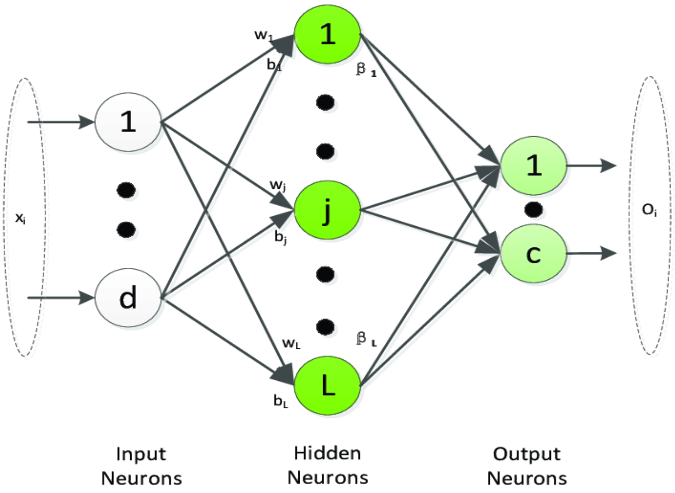


FIG2: Architecture of Extreme machine Learning(EML)

A novel learning method, known as the Extreme Learning Machine (ELM), has been designed for feed forward neural networks that only have a single hidden layer. Neural network learning algorithm: compared to the conventional technique, it resolves the sluggish training speed and over fitting issues. Empirical risk minimization theory is applied in the learning process of ELM and this method only requires a single iteration. Algorithmically, the approach avoids iterative procedures and local minimization. Because of superior generalisation ability, robustness, and controllability, and faster learning rates, it has found several applications in diverse sectors and industries. We undertake a study in this work on the recent advances in ELM's algorithm development, theory, and applications. It begins by analysing the theoretical principles and algorithm elements of ELM, and then it covers current accomplishments in ELM, such as models and applications. The paper then discusses future possibilities for ELM, including current research and development developments. Single-hidden layer feedforward neural networks (SLFNs) with a local minimum.

The algorithm conceived of ELM:

a single hidden layer feedforward neural network approach for classification and regression that's capable of making large jumps in the learning process The hidden layer nodes can be set to vary in number in order to adaptively fit the input data to the network architecture, with the least-squares approach used to generate the weights and biases for the output layer, and the training process finished by one single mathematical modification. A training pace that is three times as fast than previous BPL schemes based on gradient decline has been achieved (usually 10 times or more)

Parallel Processıng

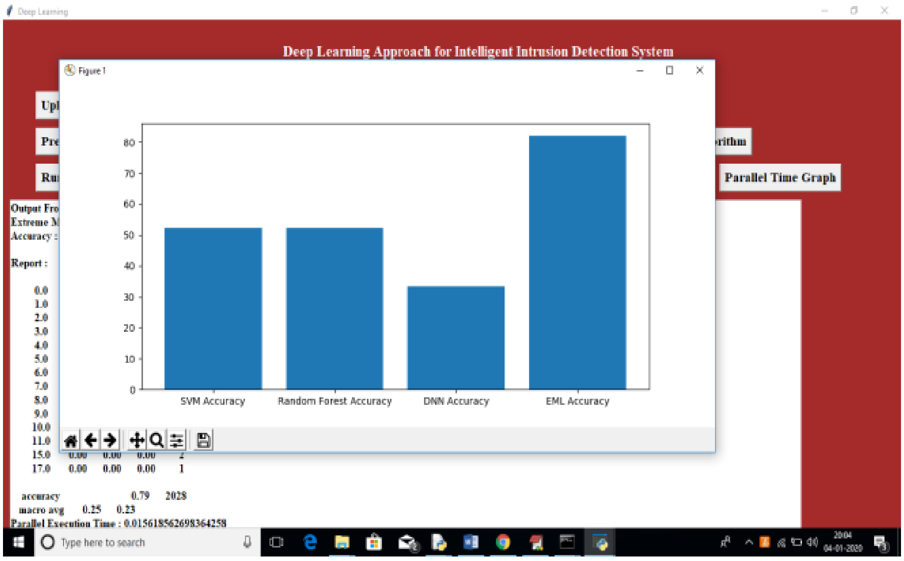
When using parallel processing, two or more processors (CPUs) are utilized to tackle various aspects of a job. Distributing a portion of a work over numerous processors will minimize the time required to complete the whole operation.

The workings of parallel processing:

A typical computer scientist will take a complex task and break it down into multiple parts, and then each part is assigned to a processor. These processors work on their individual pieces, and the data is reconstructed by a software tool so that it can be used to find the solution or carry out the task. Processors work as they typically do, doing processes in parallel as specified. Processors will also be dependent on software to help them coordinate and keep track of the changes in their data values. If the processors stay synchronised, when the work is completed, the programme will stitch all the bits of data together. It is possible to employ parallel processing even if computers have only one processor each, but they would have to be connected over a network to form a cluster.

Results

In this research we have implemented Extreme Machine Learning algorithm and parallel processing to get better prediction accuracy and less execution time.



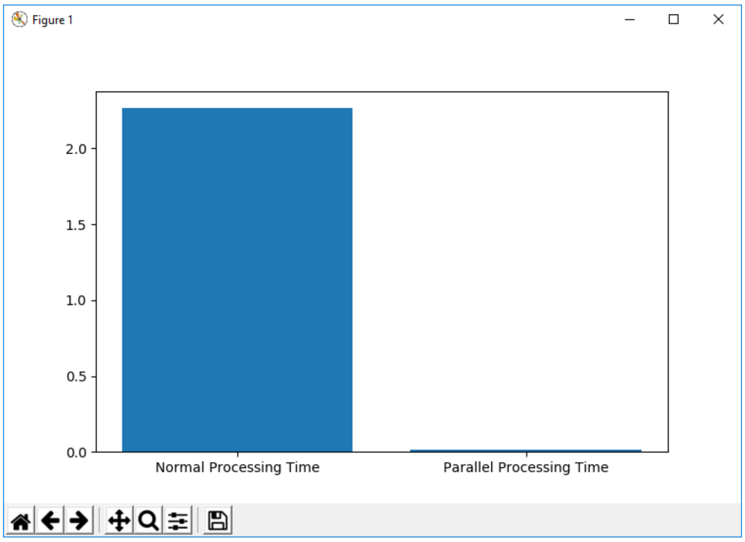
```
Output Fro
Extreme M
Accuracy :

Report :

    0.0
    1.0
    2.0
    3.0
    4.0
    5.0
    6.0
    7.0
    8.0
    9.0
   10.0
   11.0
   15.0    0.00    0.00    0.00    2
   17.0    0.00    0.00    0.00    1

  accuracy                 0.79    2028
 macro avg    0.25    0.23
Parallel Execution Time : 0.015618562698364258
```

Above figure represents the accuracy comparison graph of SVM, Random Forest, DNN and EML in which EML will give better accuracy



Above figure represents the processing time comparison with parallel and without parallel processing. In which parallel processing technique will takes less time.

Conclusıons

In this paper, we proposed a hybrid intrusion detection alert system using a highly Scalable framework which has the capability to analyze the network and host-level activities. The framework employed distributed deep learning model with DNNs and EML for handling and analyzing very large scale data in real time. The EML model was chosen by comprehensively evaluating their performance in comparison to classical machine learning classifiers on various benchmark IDS datasets. In addition, we collected host-based and network-based features in real-time and employed the proposed EML model for detecting attacks and intrusions. In all the cases, we observed that EML exceeded in performance when compared to the classical machine learning classifiers. Our proposed architecture is able to perform better than previously implemented classical machine learning algorithms.

References

[1] Mukherjee, B., Heberlein, L. T., & Levitt, K. N. (1994). Network intrusion detection. IEEE network, 8(3), 26-41.

[2] Larson, D. (2016). Distributed denial of service attacks-holding back the flood. Network Security, 2016(3), 5-7.

[3] Staudemeyer, R. C. (2015). Applying long short-term memory recurrent neural networks to intrusion detection. South African Computer Journal, 56(1), 136-154.

[4] Venkatraman, S., Alazab, M. "Use of Data Visualisation for Zero-Day Malware Detection," Security and Communication Networks, vol. 2018, Article ID 1728303, 13 pages, 2018. https://doi.org/10.1155/2018/1728303

[5] Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E. S. (2018). A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Communications Surveys & Tutorials.

[6] Azab, A., Alazab, M. & Aiash, M. (2016) "Machine Learning Based Botnet Identification Traffic" The 15th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (Trustcom 2016), Tianjin, China, 23-26 August, pp. 1788-1794.

[7] Vinayakumar R. (2019, January 19). vinayakumarr/Intrusion-detection v1 (Version v1). Zenodo. http://doi.org/10.5281/zenodo.2544036

[8] Tang, M., Alazab, M., Luo, Y., Donlon, M. (2018) Disclosure of cyber security vulnerabilities: time series modelling, International Journal of Electronic Security and Digital Forensics. Vol. 10, No.3, pp 255 - 275.

[9] V. Paxson. Bro: A system for detecting network intruders in realtime. Computer networks, vol. 31, no. 23, pp. 24352463, 1999. DOI http://dx.doi. org/10.1016/S1389-1286(99)00112-7

[10] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436. [11] Xin, Y., Kong, L., Liu, Z., Chen, Y., Li, Y., Zhu, H., ... & Wang, C. (2018). Machine Learning and Deep Learning Methods for Cybersecurity. IEEE Access.

[12] Hofmeyr, S. A., Forrest, S., & Somayaji, A. (1998). Intrusion detection using sequences of system calls. Journal of computer security, 6(3), 151180.

[13] Forrest, S., Hofmeyr, S. A., Somayaji, A., & Longstaff, T. A. (1996, May). A sense of self for unix processes. In Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on (pp. 120-128). IEEE.

[14] Hubballi, N., Biswas, S., & Nandi, S. (2011, January). Sequencegram: n-gram modeling of system calls for program based anomaly detection. In Communication Systems and Networks (COMSNETS), 2011 Third International Conference on (pp. 1-10). IEEE.

[15] Hubballi, N. (2012, January). Pairgram: Modeling frequency information of lookahead pairs for system call based anomaly detection. In Communication Systems and Networks (COMSNETS), 2012 Fourth International Conference on (pp. 1-10). IEEE.