



Machine Learning and Deep Learning approaches for Network Intrusion Detection: A Systematic Study

Ms. Kanika Kundu

Assistant Professor, Bachelor of Computer Application, Maharaja Surajmal Institute, C-4, Janak Puri

New Delhi-110058, India

Abstract: Network dimensions and its related data is increasing at a rapid rate due to the huge improvement in the area of internet and broadcasting. This has widely imposed a hindrance in accurate detection of intrusions since more and more innovative tactics are introduced. Existence of invaders with a goal to harm the network cannot be disregarded. Inspection of network traffic is done in order to fend off a system from intrusion with the help of a tool known as intrusion detection system (IDS) so that the system's confidentiality and integrity is maintained. IDS encounters various issues in achieving high detection accuracy rate and reduction of false alarm rate although several attempts have been made to improve it. Many IDS build on basis of ML and DL are being efficiently deployed as a possible remedy to detect intrusions in a system. This paper showcases the advancements of IDS with the help of a mechanism, selection of dataset and evaluation metrics. It also depicts the shortcomings of the existing IDS and highlights the scope of research for betterment of IDS.

Keywords: Network Intrusion Detection System, Firewall, Machine Learning, Cyberspace.

I. INTRODUCTION

Network security has become one of the important areas of research due to the growth of internet and broadcasting technology. It employs tools like firewall, antivirus software, and intrusion detection system (IDS) to ensure the security of the network and all its associated assets within a cyberspace [1]. Among these, network-based intrusion detection system (NIDS) is the attack detection mechanism that provides the desired security by constantly monitoring the network traffic for malicious and suspicious behavior [2][3].

The idea of IDS was first proposed by Jim Anderson in 1980 [4]. Since then, many IDS products were developed and matured to satisfy the needs of network security [5]. The size of the network is increasing tremendously with the increase in the number of applications executed by each node due to the rapid advancement in the technology. With this a large volume of data is being produced and is distributed among various nodes of the network. Generation of new vulnerabilities by either mutation or a new attack has raised a big question mark on the security of the data and nodes of network. Each node in the network is equally susceptible to security breaches. For instance, the data node may be very important for an organization. Data is very important for an organization and any harm done to it may put the reputation of the organization at stake and may result into losses. Existing IDSs have shown inefficiency in detecting various attacks including zero-day attacks and reducing the false alarm rates (FAR) [6]. Therefore, there is an alarming need for an efficient and cost-effective IDS in order to keep our data free from alterations.

Researchers have now investigated ML and DL techniques in order to create robust IDS. Both ML and DL come under the big umbrella of artificial intelligence (AI) and aim at learning useful information from the big data [7]. These techniques have gained enormous popularity in the field of network security, over the last decade due to the invention of very powerful graphics processor units (GPUs) [8]. Learning characteristics of network traffic and then predicting the abnormal activities can be easily done by ML and DL tools. The ML-based IDS depends heavily on feature engineering to learn useful information from the network traffic [9]. While DL-based IDS do not rely on feature engineering and are good at automatically learning complex features from the raw data due to its deep structure [10].



II. IDS: CONCEPT AND CLASSIFICATION

A. CONCEPTION

IDS comprises of “intrusion” and “detection system.” *Intrusion* refers to an unauthorized access to the information within a computer or network systems to compromise its integrity, confidentiality, or, availability [12][13]. Illegal activities can be identified with the help of detection systems. So, *IDS* is a security tool that constantly monitors the host and network traffic to detect any suspicious behavior that violates the security policy and compromises its confidentiality, integrity, and availability [11][14]. Alerts will be generated regarding the malicious threat being detected to the system administrators by the IDS. Figure 1 showcases submissive implementation of NIDS, where it is attached to a switch which is arranged with the switch port analyzer. The task is to mirror all the incoming and outgoing network traffic to NIDS for performing traffic monitoring to detect intrusions. NIDS can also be deployed in between the firewall and the network switch to allow all the traffic to pass through NIDS [2].

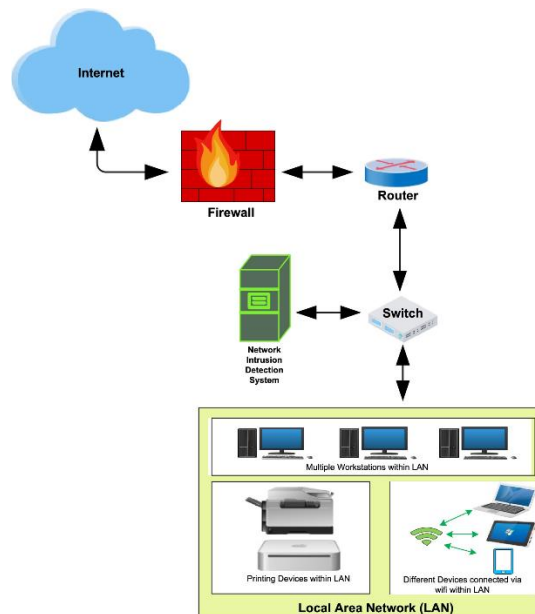


Figure 1: Local Area Network and NIDS

B. Classification of IDS

IDS can be classified with the perspective of its deployment or detection methods. A classification taxonomy is given in Figure below.

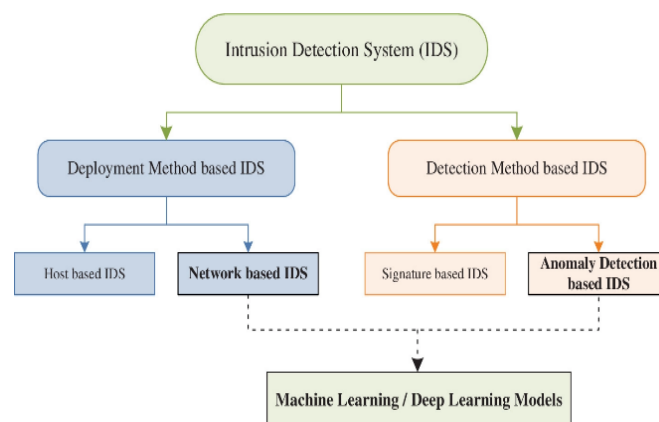


Figure 2: Intrusion Detection System

III. AI METHODS FOR NIDS

This section provides a general methodology of the AI-based NIDS along with the details of the most commonly used ML and DL algorithms used to design an efficient NIDS. Both ML and DL are broadly classified as supervised and

unsupervised algorithms. In *supervised algorithms*, the useful information is extracted from the labeled data. While *unsupervised algorithms* rely on the unlabeled data to extract useful features and information [15].

A. A general AI-based NIDS methodology

A NIDS developed using ML and DL methods usually involves following three major steps as depicted in Figure, that is, (i) Data preprocessing phase, (ii) Training phase, and (iii) Testing phase. For all the proposed solutions, the dataset is first preprocessed to transform it into the format suitable to be used by the algorithm. This stage typically involves encoding and normalization. Sometimes, the dataset requires cleaning in terms of removing entries with missing data and duplicate entries, which is also performed during this phase. The preprocessed data is then divided randomly into two portions, the *training dataset*, and the *testing dataset*. Typically, the training dataset comprises almost 80% of the original dataset size and the remaining 20% forms testing dataset [16][17]. The ML or DL algorithm is then trained using the training dataset in the training phase. The time taken by the algorithm in learning depends upon the size of the dataset and the complexity of the proposed model. Normally, the training time for the DL models requires more training time due to its deep and complex structure. Once the model is trained, it is tested using the testing dataset and evaluated based on the predictions it made. In the case of NIDS models, the network traffic instance will be predicted to belong to either benign (normal) or attack class.

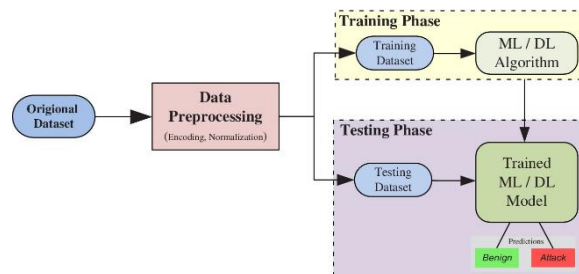


Figure 3: NIDS Model

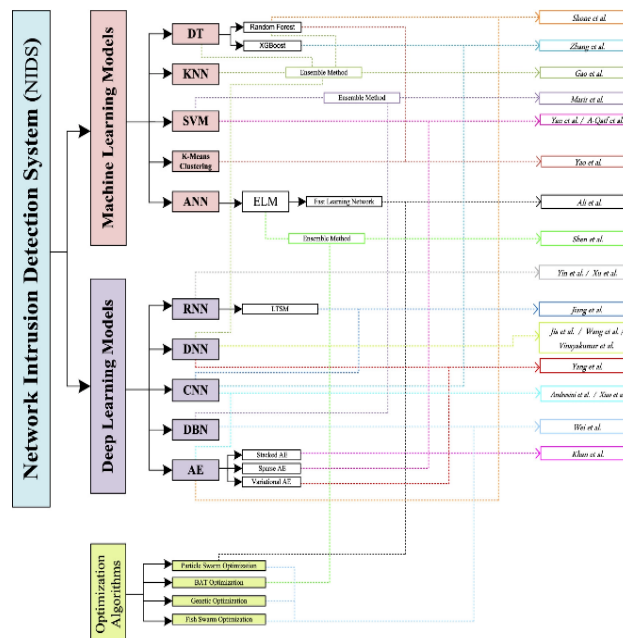


Figure 4: Taxonomy of recent ML- and DL-based techniques used for NIDS.

B. ML Algorithms

ML is a subset of AI that includes all the methods and algorithms which enable the machines to learn automatically using mathematical models in order to extract useful information from the large datasets.

- Decision Tree
- K-Nearest Neighbor
- Support Vector Machine
- K-Mean Clustering



- Artificial Neural Network

C. Deep-Learning Algorithms

- Recurrent Neural Networks
- Auto Encoder
- Deep Neural Network
- Deep Belief Network

Study	Algorithm		Methodology
	ML	DL	
Yin et al ⁹¹		✓	Recurrent Neural Network
Shen et al ⁸¹	✓		Ensemble Method with Optimization using BAT algorithm
Shone et al ⁹⁷	✓	✓	Non Symmetric Deep Auto Encoder with Random Forest
Ali et al ⁷⁹	✓		Fast Learning Network and Particle Swarm Algorithm
Jia et al ¹⁰⁹		✓	Deep Neural Network
Wang et al ¹¹¹		✓	Deep Neural Network
Yan et al ⁶⁷	✓	✓	Sparse Auto Encoder with SVM
Naseer et al ⁹⁴	✓	✓	Comparison between different ML and DL-based IDS Models
Xu et al ⁹³		✓	Neural Network using gated Recurrent Units as memory with Multiple Layer Perception.

Figure 5: Methodologies

IV. EVALUATION METRICS

This section explains the most commonly used evaluation metrics for measuring the performance of ML and DL methods for IDS. All the evaluation metrics are based on the different attributes used in the *Confusion Matrix*, which is a two-dimensional matrix providing information about the Actual and Predicted class and includes:

1. *True Positive (TP)*: The data instances correctly predicted as an Attack by the classifier.
2. *False Negative (FN)*: The data instances wrongly predicted as Normal instances.
3. *False Positive (FP)*: The data instances wrongly classified as an Attack.
4. *True Negative (TN)*: The instances correctly classified as Normal instances.

The diagonal of confusion matrix denotes the correct predictions while nondiagonal elements are the wrong predictions of a certain classifier. Further, the different evaluation metrics used in the recent studies are,

- **Precision**: It is the ratio of correctly predicted Attacks to all the samples predicted as Attacks.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall**: It is a ratio of all samples correctly classified as Attacks to all the samples that are actually Attacks. It is also called a Detection Rate.

$$\text{Recall} = \text{Detection Rate} = \frac{TP}{TP + FN}$$

- **False alarm rate**: It is also called the false positive rate and is defined as the ratio of wrongly predicted Attack samples to all the samples that are Normal.



$$\text{False Alarm Rate} = \frac{FP}{FP + TN}.$$

- **True negative rate:** It is defined as the ratio of the number of correctly classified Normal samples to all the samples that are Normal.

$$\text{True Negative Rate} = \frac{TN}{TN + FP}.$$

- **Accuracy:** It is the ratio of correctly classified instances to the total number of instances. It is also called as Detection Accuracy and is a useful performance measure only when a dataset is balanced.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}.$$

- **F-Measure:** It is defined as the harmonic mean of the Precision and Recall. In other words, it is a statistical technique for examining the accuracy of a system by considering both precision and recall of the system.

$$F \text{ Measure} = 2 \left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right).$$

V. OBSERVATIONS

The efficiency of the AI-powered NIDS highly depends upon training using a suitable dataset. For ML models, the algorithm can be trained using a small dataset to achieve better results. But in case of the larger dataset, then ML is not suitable unless the dataset is labeled in nature. Since labeling is an expensive and time-consuming process, DL methods are preferred for larger datasets. These methods will learn and extract useful patterns from raw datasets. To make NIDS efficient in detecting zero-day attacks, it needs to be trained regularly with the new data obtained as a result of monitoring the network traffic. The large dataset and deep nature of the DL algorithms will make the learning process more resource hungry in terms of computational resources and time consumption. The more the NIDS model is trained, the more efficiently it will detect intrusions.

Based on the reviewed article, it is observed that during the past three years, the researchers focused on DL tools for designing the IDS system as shown in Figure. It is noticed that 60% of the proposed methods are purely based on the DL approaches, 20% solutions use hybrid approach involving the combination of ML- and DL-based algorithms while only 20% proposed solutions are based on ML methods. As discussed earlier, DL models are complex in nature and require extensive computational resources. This is made possible due to the advent of the GPU and has contributed to a vast increase in the use of DL-based algorithms in the design of IDS.

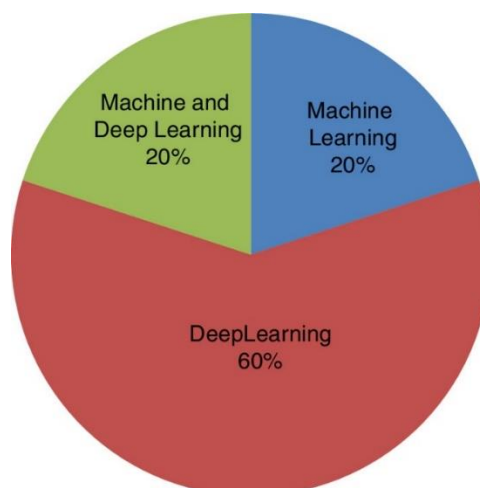


Figure 6: Usage Percentage



The analysis of the performance metrics used by the researchers for the evaluation of the methodology is shown in Figure. It is noted that the most widely used performance metrics are Detection Accuracy and Recall (Detection rate). It is obvious that for efficient network security, the IDS demands a higher Accuracy and Detection rate. So, to investigate the efficiency and effectiveness of the proposed methodology, these two-performance metrics should be considered. For a typical IDS designed using ML / DL tools, the Accuracy, Recall, and F-measure should be the compulsory performance metric besides others to show its ability in detecting intrusions.

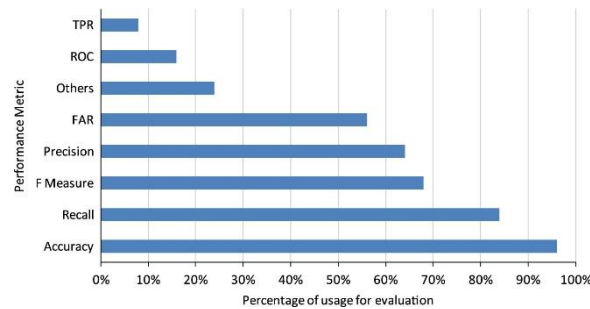


Figure 7: Parameters

VI. CONCLUSION

The study also shows that 60% of the proposed methodologies were tested using KDD Cup'99 and NSL-KDD datasets mainly because of the availability of extensive results using these datasets. But these datasets are quite old to address modern network attacks, and hence limits the performance of the proposed methodologies in real-time environments. For AI-based NIDS methods, the model should be tested with the latest updated dataset like CSE-CIC-IDS2018 for better performance in terms of detection accuracy for intrusions. It highlights the research gaps in improving the model performance for low-frequency attacks in a real-world environment and to find efficient solutions to reduce complexity for the proposed models. Proposing an efficient NIDS framework using less complex DL algorithms and have an effective detection mechanism is a potential future scope of research in this area.

REFERENCES

- [1]. Tarter A. Importance of cyber security. Community Policing-A European Perspective: Strategies, Best Practices and Guidelines. New York, NY: Springer; 2017: 213- 230, DOI: https://doi.org/10.1007/978-3-319-53396-4_15.
- [2]. Li J, Qu Y, Chao F, Shum HP, Ho ES, Yang L. Machine learning algorithms for network intrusion detection. AI in Cybersecurity. New York, NY: Springer; 2019: 151- 179, DOI: https://doi.org/10.1007/978-3-319-98842-9_6.
- [3]. Lunt TF. A survey of intrusion detection techniques. Computer and Sec. 1993; 12(4): 405- 418. [https://doi.org/10.1016/0167-4048\(93\)90029-5](https://doi.org/10.1016/0167-4048(93)90029-5).
- [4]. Anderson JP. Computer Security Threat Monitoring and Surveillance. Fort Washington, PA: James P Anderson Co; 1980.
- [5]. Debar H, Dacier M, Wespi A. Towards a taxonomy of intrusion-detection systems. Comput Netw. 1999; 31(8): 805- 822. [https://doi.org/10.1016/S1389-1286\(98\)00017-6](https://doi.org/10.1016/S1389-1286(98)00017-6).
- [6]. Hoque MS, Mukit M, Bikas MA. An implementation of intrusion detection system using genetic algorithm. *International Journal of Network Security & Its Applications*. 2012;4(2):109–120. doi: 10.5121/ijnsa.2012.4208.
- [7]. Prasad R, Rohokale V. Artificial intelligence and machine learning in cyber security. In: *Cyber Security: The Lifeline of Information and Communication Technology*. Springer; 2020:231–247. doi: 10.1007/978-3-030-31703-4_16.
- [8]. Lew J, Shah DA, Pati S, Cattell S, Zhang M, Sandhupatla A, et al. Analyzing machine learning workloads using a detailed GPU simulator. In: *Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE; 2019:151–152. doi: 10.1109/ISPASS.2019.00028.
- [9]. Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E. Deep learning applications and challenges in big data analytics. J Big Data. 2015; 2(1): 1. <https://doi.org/10.1186/s40537-014-0007-7>.
- [10]. Dong B, Wang X. Comparison deep learning method to traditional methods using for network intrusion detection. In: *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)*. IEEE; 2016:581–585. doi:10.1109/ICCSN.2016.7586590
- [11]. Vasilomanolakis E, Karuppayah S, Mühlhäuser M, Fischer M. Taxonomy and survey of collaborative intrusion detection. ACM Comput Surv. 2015; 47(4): 1- 33. <https://doi.org/10.1145/2716260>



- [12]. Mukkamala S, Janoski G, Sung AH. Intrusion detection using neural networks and support vector machines. In: *Proceedings of the 2002 International Joint Conference on Neural Networks (IJCNN'02)*. Honolulu, HI, USA: IEEE; 2002;2:1702–1707. doi:10.1109/IJCNN.2002.1007774.
- [13]. Garcia-Teodoro P, Diaz-Verdejo J, Maciá-Fernández G, Vázquez E. Anomaly-based network intrusion detection: techniques systems and challenges. *Computer Secur.* 2009; 28(12): 18- 28. <https://doi.org/10.1016/j.cose.2008.08.003>
- [14]. Denning DE. An intrusion-detection model. *IEEE Trans SoftwEng.* 1987; 2: 222- 232. <https://doi.org/10.1109/TSE.1987.232894>.
- [15]. Zanero S, Serazzi G. Unsupervised learning algorithms for intrusion detection. In: *Proceedings of the 2008 IEEE Network Operations and Management Symposium (NOMS 2008)*. Salvador, Bahia, Brazil: IEEE; 2008:1043–1048. doi:10.1109/NOMS.2008.4575276.
- [16]. Imamverdiyev Y, Abdullayeva F. Deep learning method for denial-of-service attack detection based on restricted Boltzmann machine. *Big Data.* 2018; 6(2): 159- 169. <https://doi.org/10.1089/big.2018.0023>.
- [17]. Alsughayyir B, Qamar AM, Khan R. Developing a network attack detection system using deep learning. In: *2019 International Conference on Computer and Information Sciences (ICCIS)*. Sakaka, Saudi Arabia: IEEE; 2019:1–5. doi:10.1109/ICCISci.2019.8716389.