Network Intrusion Detection Using Deep Belief Network (DBN)

Wael Farouk Elsersy
Faculty of Computer Science.
October University for
Modern Sciences and Arts (MSA)
Giza, Egypt
wfarouk@msa.edu.eg 0000-0002-7501-6889

Moataz Samy
Faculty of Computer Science.
October University for
Modern Sciences and Arts (MSA)
Giza, Egypt
moasamy@msa.edu.eg

Ahmed ElShamy
Faculty of Computer Science.
October University for
Modern Sciences and Arts (MSA)
Giza, Egypt
aelshamy@msa.edu.eg

2024 Intelligent Methods, Systems, and Applications (IMSA) | 979-8-3503-6263-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/IMSA61967.2024.10652732

***Abstract*—The escalating Interconnectedness of devices has given rise to a surge in cyber-security threats, particularly zero-day attacks, which pose a significant challenge to traditional Intrusion Detection Systems (IDSs). This paper delves into the realm of behavior-based IDSs, leveraging Deep Neural Networks (DNNs), with a specific focus on the detection of network attacks. The effectiveness of these systems is closely tied to the quality of the training Dataset, where underrepresented samples can compromise detection performance, Our research centers on the development and evaluation of Deep Belief Networks (DBNs) as a robust solution for detecting cyber-attacks within networks of connected devices. The cornerstone of our investigation is the utilization of the CICIDS2017 Datasets, a comprehensive repository that facilitates both training and evaluation of our proposed DBN approach. To address the challenges posed by imbalanced Datasets, we employ various class balancing techniques, systematically evaluating their impact on detection performance. In the context of network attacks, our study pays special attention to notorious threats such as SQL injection, Cross-Site Scripting (XSS), Denial of Services (DoS), Distributed Denial of Services (DDoS), and other malicious activities. These attacks, often veiled in the subtleties of network traffic, demand sophisticated detection mechanisms. The empirical evaluation involves a comparative analysis with a conventional Multi-Layer Perceptron (MLP) model and the prevailing state-of-the-art IDSs. Our proposed DBN approach demonstrates competitive and promising results, showcasing significant performance improvements in the detection of attacks that are underrepresented in the training Datasets. The findings underscore the efficacy of DBNs in tackling the intricate challenges posed by emerging and sophisticated cyber threats. This research not only contributes to the advancement of intrusion detection methodologies but also provides insights into addressing the dynamic landscape of cyber threats, emphasizing the need for adaptive and intelligent systems in safeguarding networked environments.**

***Index Terms*—Intrusion Detection, DDoS, XSS, CICIDS2017, Machine Learning, DOS**

I. Introduction

The ever-expanding geography of connected devices has steered in a period of unequaled convenience and effectiveness. still, this connected web has also given rise to a redoubtable array of cyber-security challenges, with zero-day attacks arising as pervasive trouble. Traditional Intrusion Discovery Systems(IDSs), counting on Deep Neural Networks(DNNs), stand at the van of defense against these sophisticated attacks. The efficacy of these systems, crucially dependent on the quality of their training Datasets, faces a redoubtable challenge in the form of underrepresented samples, which can compromise discovery performance.

This research endeavors to unravel the intricacies of behavior-based IDSs, specifically focusing on the detection of network attacks. In this context, the study introduces the utilization of Deep Belief Networks (DBNs) as a formidable solution, offering a promising avenue for enhancing Cybersecurity within networks of connected devices. The CICIDS2017 Dataset serves as the cornerstone of our investigation, providing a comprehensive platform for training and evaluating the proposed DBN approach, Addressing the critical issue of imbalanced Datasets, we employ a series of class balancing techniques, systematically assessing their impact on detection performance. As we navigate the intricate landscape of cyber threats, our study zooms in on notable menaces, including SQL Injection, Cross-Site Scripting (XSS), Denial of Service (DoS), Distributed Denial of Service (DDoS), and other malicious activities that lurk within the nuances of network traffic, We can Go through Each Attack to Introduce a Simple Idea what are the Attacks We Defending and Detecting In CICIDS2017 database we Used[1]:

Cross-site scripting (XSS) Attacks: emerged as a pervasive and potent vulnerability, leaving digital geographies susceptible to exploitation. XSS attacks, characterized by the injection of vicious scripts into web runners, take advantage of the trust placed in putative inoffensive websites. These scripts latterly execute within the user's cybersurfer, initiating a waterfall of consequences that hazard both data integrity and user's confidentiality, In the intricate cotillion between druggies and web runners[2], adversaries strategically fit dangerous scripts, frequently using vulnerabilities in input confirmation processes. Once bedded, these scripts apply significant power, enabling bushwhackers to clandestinely access sensitive information, manipulate the content of web runners, and indeed execute conduct on behalf of compromised druggies. The multifaceted nature of XSS attacks elevates the stakes, posing substantial pitfalls to the security and sequestration of individualities and

979-8-3503-6263-3/24/$31.00 ©2024 IEEE

associations.

SQL Injection: SQL Injections vulnerabilities pose a critical and pervasive threat to web applications, earning recognition as one of the most serious risks in the realm of Cybersecurity. These vulnerabilities enable attackers to manipulate user-provided data in such a way that it becomes interpreted as SQL code, allowing unauthorized access to underlying databases. The implications of successful SQL injection attacks are severe, ranging from potential identity theft and loss of confidential information to outright system compromise and corruption. A study by the Gartner Group revealed the widespread prevalence of web applications vulnerable to SQL injection, underlining the urgency of addressing this security loophole.[3] While developers have proposed coding guidelines to mitigate these vulnerabilities, the human-centering nature of their application and the challenges of rectifying legacy code make it a complex and labor-intensive task. Despite increased attention to SQL injection vulnerabilities, many proposed solutions often fall short, addressing only specific subsets of potential attacks.[4] In this research, we recognize the need for a comprehensive understanding of SQL injection attacks, their variations, and the effectiveness of current detection and prevention techniques, thereby contributing to the advancement of intrusion detection systems in safeguarding against this persistent threat.

Denial of Service (DoS) / Distributed Denial of Service (DDoS): Denial of Service(DoS) attacks, particularly Distributed Denial of Service(DDoS) attacks, have become decreasingly common in the cyberspace realm, posing a heightened threat to computer and network services. Accordingly, associations and individualities are investing in strategic plans to fortify their serviceability and services against cyber pitfalls, including DDoS attacks, The DDoS involves overwhelming a target system with an expansive volume of requests, orchestrated by multiple accommodated hosts to disrupt and damage the coffers of the victimized hosts. Unlike traditional attacks that exploit specific vulnerabilities, DDoS attacks induce detriment by coinciding multitudinous hosts to drown the victim's machine contemporaneously. While colorful discovery styles live, there's no reliable approach to both detecting and precluding DDoS attacks. Accordingly,[5] baffling DDoS attacks remain a redoubtable challenge, challenging the capability to separate between licit and vicious business. DDoS poses substantial trouble to the vacuity of cyberspace serviceability, causing dislocations to individual hosts, major marketable realities, and indeed critical structure services. The fiscal ramifications are significant, with DDoS attacks potentially going victim millions of bones due to dragged attainability. especially, the prosecution of DDoS tools requires minimum specialized moxie, rendering them easy to launch and challenging to describe.

II. Related Works

Computer security monitoring in widespread networks is a recent and open issue of research. The exchange of security information among organizations differs from that applied by the traditional intrusion detection systems In the Golden Age of Artificial Intelligence and Machine Learning, Using Machine Learning Models in IDs Now Has Become a Basic Requirement for the Network Security Monitoring Systems to Be Qualified to Be Used.

Machine Learning Intrusion Detection Systems That are Used Now are devices that are linked to the gateway and when the Packets arrive, They extract the Payload from the Packets and start making decisions based on The Payload or Source and Destination and IP address and the Port number Found in The Packet.[6] Machine Learning Methodologies Used for the IDS Usually Depending on the SVM and we Can find This in the Following Researches[7][8] On what We Conclude From their Works that these Implementation Methodologies That at high Network Traffic, it takes too much longer time to classify the Unseen data. [9]

III. Methodology

Our Methodology in This Paper Is to Implement a Deep Belief Network Based Intrusion Detection System (IDS).

A. Model Training

The Implemented Model Which is a Deep Belief Network, requires Two Steps In training, We Will Discuss the Process in the Next Subsections.

1) Unsupervised Pre-training: In the first phase of training the Deep Belief Network (DBN), unsupervised pre-training is employed. This phase involves training each layer of the network separately using Restricted Boltzmann Machines (RBMs). The process can be summarized as follows:

1) RBM Training for Each Layer:
    - Each RBM is trained to reconstruct its input.
    - The weights of the RBM are adjusted during training.
    - The input layer of each RBM is fed the input data, and the RBM learns to reconstruct this input.
2) Layer-by-Layer Pre-training:
    - The pre-training process is conducted in a layer-by-layer fashion.
    - Greedy learning is employed, where each layer is pre-trained independently of the others.
3) Repeating Until All Layers Are Pre-trained:
    - The above process is repeated until all RBM layers in the DBN are pre-trained.
    - This phase aims to capture high-dimensional representations of the data in an unsupervised manner.

2) Supervised Fine-tuning: After the unsupervised pre-training, the network undergoes supervised fine-tuning. This phase involves optimizing all the weights of the network using supervised learning techniques. Key steps in this phase include:

1) Optimizing Weights:
    - All the weights of the DBN are optimized during fine-tuning.

- Stochastic gradient descent (SGD) is typically used as the optimization algorithm.

2) Back-Propagation:
   - Back-propagation is employed to adjust the weights based on the error in the network's predictions compared to the actual labels.
   - This is done in a supervised manner, using labeled data to guide the optimization process.

3) Performance Metrics:

1) F1 Score: Primary metric, as it balances precision and recall, providing a comprehensive evaluation considering both false positives and false negatives.
2) Precision: assess the model's ability to correctly identify positive instances and capture all relevant positive instances, respectively.
3) Recall: assess the model's ability to correctly identify positive instances and capture all relevant positive instances, respectively.

IV. Datasets

In The Dataset We Use CICIDS2017, which contains samples of different types of attacks.

TABLE I: CICIDS 2017 Dataset

Attack Type	Attack Definition	Attack Count
Benign	Default Packets don't contain Malicious Code	1,807,787
DoS/DDoS	Samples Specified For DoS/DDoS Attacks	320,269
Port-Scan	Samples Specified For Web Attacks Responsible for Port-Scan attacks	8,551
Brute Force	Samples Specified For Web Attacks Responsible for Brute Force attacks	8,551
Web Attacks	Samples Specified For Web Attacks Responsible for XSS and SQL Injection	2,118
Botnet	Samples Specified For Web Attacks Responsible for Botnet attacks	1,943

As Table I shows there are 6 main labels that represent the types of attacks that are expected to be found in the dataset along with their definition and count

V. DBN and MLP Architecture

In this study, we implemented and tuned two different deep learning classifiers,the architecture of the implemented MLP comprises multiple fully connected layers, with 49 nodes used in the input layer to represent the number of input features. After fine-tuning, two hidden layers with 64 nodes each were set, and the ReLU activation function was applied in the hidden layers. The output layer consists of six nodes, each representing one class. The Softmax function is used in the output layer for multi-class classification.

the architecture of the implemented DBN. After fine-tuning, five RBMs are stacked with (49, 128), (128, 256), (256, 128), (128, 128), and (128, 64) visible/hidden nodes set per RBM, respectively. The output from the last RBM is connected to a fully connected layer with 6 nodes for multi-class classification using the Softmax function. The training parameters of the DBN and MLP are shown in Table II and Table III

TABLE II: DBN Pre-training and Fine-tuning Parameters

Parameter	Pre-training	Fine-tuning
Epochs	10	30
Learning rate	0.1	0.001
Batch size	64	128
Momentum	0.9	-
Optimizer	SGD	Adam
Loss function	-	Cross-entropy
Gibbs step	1 step	-
Weight init.	Xavier initialiser	-
Bias init.	Zeros (0)	-

TABLE III: MLP Parameters

Parameter	Value
Epochs	10
Learning rate	0.02
Batch size	64
Momentum	0.9
Weight decay	-
Optimizer	SGD
Loss function	Cross-entropy

VI. Results

We'll be discussing the results of both our models in this section starting with the DBN.

A. DBN Results

The DBN IDS showed promising results when it came to it's ROC outperforming the MLP.

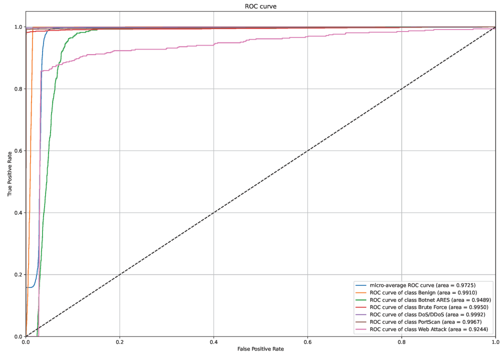

Fig. 1: Deep Belief ROC

As Figure1 shows the Area Under Curve(AUC) of the micro-average curve shows really promising results being 0.972 which means that the DBN could successfully classify the attacks with accuracy and precision.

However while training the model some unexpected results were found as demonstrated in the figures below

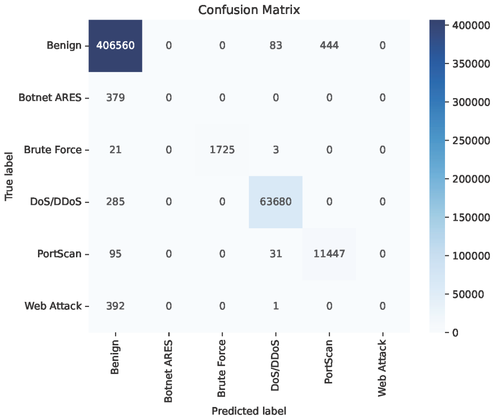

Fig. 2: Deep Belief Confusion Matrix

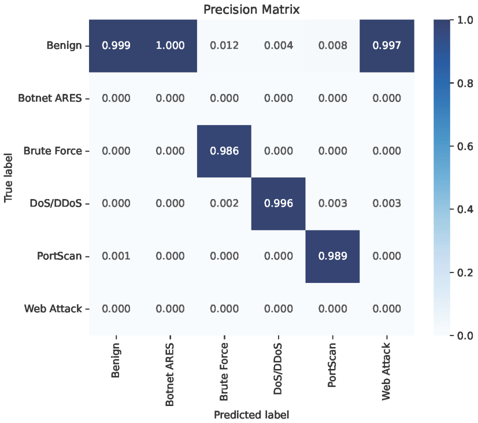

Fig. 3: Deep Belief Precision Matrix

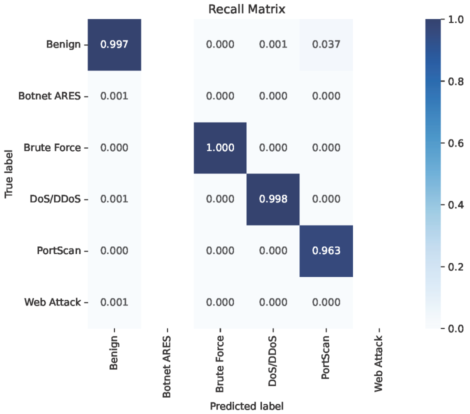

Fig. 4: Deep Belief Recall Matrix

Figures 2, 3, and 4 illustrate the confusion matrix, Precision matrix and Recall matrix achieved by the proposed DBN-based multi-class classification, As the results show the DBN predicts 4 Labels out of the presented 6 successfully as the botnet's and web attacks are not identified correctly but the precision of detection of other attacks are high with DoS/DDoS and portScan having precision of 0.996 and 0.991 respectively, the model shows Bias towards Benign packets as most miss-identified attacks are identified as Benign.

B. MLP Results

the ROC curve of the MLP while having overall lower AUC its testing results were surprising.

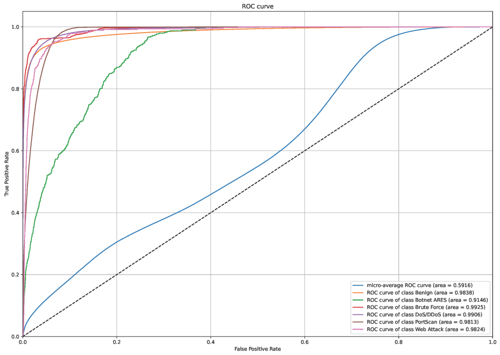

Fig. 5: Multlayer Perceptron ROC

Despite the low over AUC in the MLP as demonstrated in Figure5 its confusion matrix/precision and recall show better results than DPN in an unbalanced dataset.

The multi layer perceptron on the other hand while not completely precise in detecting Botnet attacks generally does a much better job at identifying attacks than the DBN as it identifies all the 6 labels with good precision and recall as shown in Figures 6, 7, and 8

Both models demonstrate high accuracy in classifying network traffic samples. However,it's notable that MLP works better in identifying all kinds of threats compared to our DBN model.

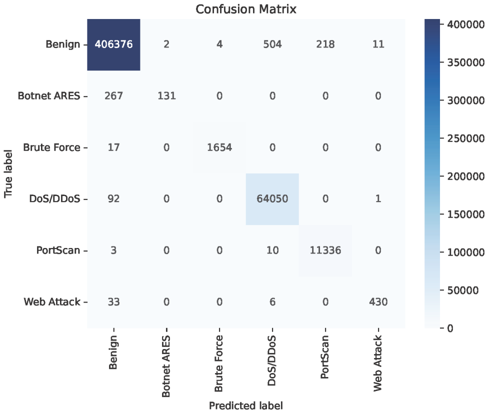

Fig. 6: Multilayer Perceptron Confusion Matrix

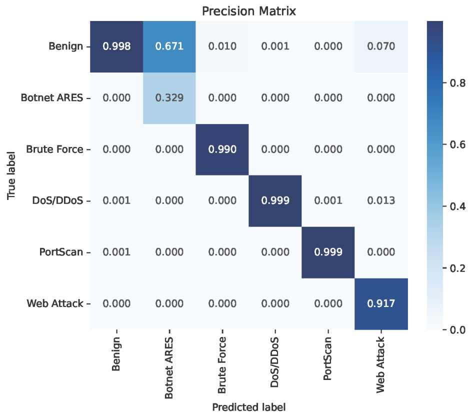

Fig. 7: Multilayer Perceptron Precision Matrix

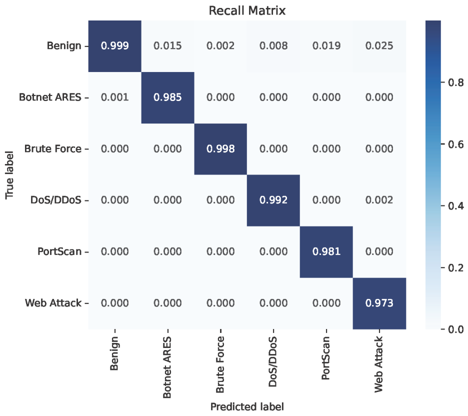

Fig. 8: Multilayer Perceptron Recall Matrix

Conclusions

To conclude our research we'd like to comment on the results our machine learning models achieved whether its a comparison between the MLP and DBN that we created or other approaches from different research papers.

C. Comparison Between Our DBN and MLP

as the results have shown for both our DBN and MLP models, while the ROC Area Under Curve for the DBN shows more promise overall than the MLP the testing results without applying any balancing techniques show that the MLP is better in the sense that it detects all the attacks and identifies them with higher accuracy while the DBN has trouble identifying Botnet and WebAttacks precisely.

D. Comparison With Other Models

As much as our DBN based IDS performance was underwhelming the results of our MLP based IDS were slightly better than that of other research paper as demonstrated in this subsection. Figure 9 illustrates the F1-Score, Precision, and Recall achieved by our machine learning methods and other methods implemented in [10],[11] and [12]. And our DBN shows low F1 Score and recall and precision but the MLP has higher overall scores that then MLP implimented in [11]'s research paper

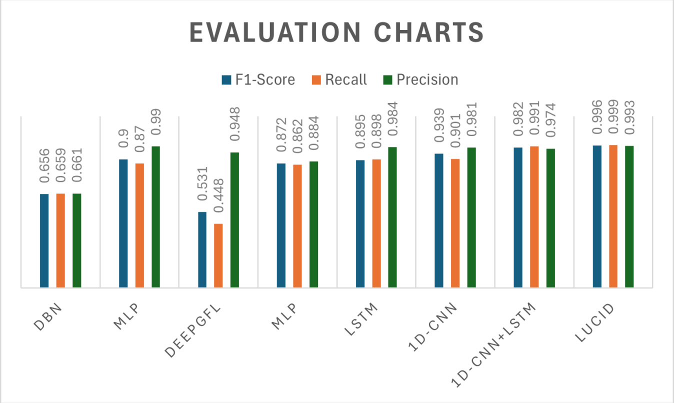

Fig. 9: Illustration of the F1-Score, Precision, and Recall.

E. Future work

In this study, we investigated the application of Deep Belief Networks (DBNs) for Network Intrusion Detection Systems (NIDS). Our exploration led to the development of two NIDS, one based on DBNs and the other on Multi-Layer Perceptrons (MLPs).

The MLP-based and DBN-based NIDS proposed in our research achieved an F1-score of 90% and 65.6%, respectively. Our experimental findings have shown slightly better result using MLPs in the classification of network intrusions than that proposed in [11].

Our recommendations for future work would be to implement the dataset balancing techniques and retraining the model and monitoring the performance, as it's expected to show some improvement in the DBN model.

References

[1] O. Belarbi, A. Khan, P. Carnelli, and T. Spyridopoulos, "An intrusion detection system based on deep belief networks," in *International Conference on Science of Cyber Security*, Springer, 2022, pp. 377–392.

[2] S. J. Weamie, "Cross-site scripting attacks and defensive techniques: A comprehensive survey," *International Journal of Communications, Network and System Sciences*, vol. 15, no. 8, pp. 126–148, 2022.

[3] W. G. Halfond, J. Viegas, A. Orso, *et al.*, "A classification of sql-injection attacks and countermeasures," in *Proceedings of the IEEE international symposium on secure software engineering*, IEEE, vol. 1, 2006, pp. 13–15.

[4] K. Wei, M. Muthuprasanna, and S. Kothari, "Preventing sql injection attacks in stored procedures," in *Australian Software Engineering Conference (ASWEC'06)*, IEEE, 2006, 8–pp.

[5] B. Nagpal, P. Sharma, N. Chauhan, and A. Panesar, "Ddos tools: Classification, analysis and comparison," in *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, IEEE, 2015, pp. 342–346.

[6] V. Matyas and J. Kur, "Conflicts between intrusion detection and privacy mechanisms for wireless sensor networks," *IEEE Security & Privacy*, vol. 11, no. 5, pp. 73–76, 2013.

[7] M. Mohammadi *et al.*, "A comprehensive survey and taxonomy of the svm-based intrusion detection systems," *Journal of Network and Computer Applications*, vol. 178, p. 102 983, 2021.

[8] H. Gharaee and H. Hosseinvand, "A new feature selection ids based on genetic algorithm and svm," in *2016 8th International Symposium on Telecommunications (IST)*, IEEE, 2016, pp. 139–144.

[9] E. Anthi, L. Williams, M. Słowińska, G. Theodorakopoulos, and P. Burnap, "A supervised intrusion detection system for smart home iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 9042–9053, 2019.

[10] Y. Yao, L. Su, and Z. Lu, "Deepgfl: Deep feature learning via graph for attack detection on flow-based network traffic," in *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*, IEEE, 2018, pp. 579–584.

[11] M. Roopak, G. Y. Tian, and J. Chambers, "Deep learning models for cyber security in iot networks," in *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)*, IEEE, 2019, pp. 0452–0457.

[12] R. Doriguzzi-Corin, S. Millar, S. Scott-Hayward, J. Martinez-del Rincon, and D. Siracusa, "Lucid: A practical, lightweight deep learning solution for ddos attack detection," *IEEE Transactions on Network and Service Management*, vol. 17, no. 2, pp. 876–889, 2020.