A Flexible Hybrid Behavior Dynamic Methodology for Network Intrusion Detection using CNN

P. JAYALAKSHMI, VISWANATH VS, SURENDHARAN TG, NAVEEN PK
[1]SRM INSTITUE OF SCIENCE AND TECHNOLOGY
[2]Department of Computer Science Engineering

ABSTRACT Artificial Intelligence (AI) has significantly impacted the development of Network Intrusion Detection Systems (NIDS). AI algorithms, such as machine learning, have enabled NID systems to analyze vast amounts of data, identify patterns, and detect anomalies in real-time. This helps in detecting and preventing cyber-attacks, which are becoming increasingly sophisticated and challenging to detect. NID systems that use AI algorithms are capable of learning from past experiences and can adapt to new threats, making them more effective in detecting intrusions. Furthermore, AI algorithms also help in reducing false positive alarms, which can be a significant hindrance in the effective functioning of NID systems. In this paper, we apply the CNN machine learning model to analyze and enhance NID performance

INDEX TERMS Artificial Intelligence, Network Intrusion Detection Systems, Machine learning

I. INTRODUCTION

Networks have taken on a crucial role in today's linked world and in contemporary commercial operations. Yet, because of their dependency on networks, they have become a popular target for hackers. Businesses are susceptible to serious harm from cyberattacks, including financial loss, harm to their reputation, and loss of critical data. Organizations must thus put strong security measures in place to shield their networks from online threats.[1]

One of the key security measures is the implementation of an intrusion detection system (IDS). Security technologies called IDSs monitor network traffic for any signs of criminal or suspicious behavior. They have the ability to detect a variety of threats, including malware, denial-of-service attacks, and port scans. IDS can produce real-time notifications that assist administrators in responding to security occurrences and taking the necessary steps to reduce the risk of harm. The alerts may also be used to create statistics and reports on the network's security, assisting managers in determining which areas need more attention.[4]

Unified Network Intrusion Detection UNID is necessary since traditional IDSs are constrained. In order to recognize certain types of attacks, conventional IDS frequently rely on a preset set of rules or signatures. As a result, they can be less able to see new or unexpected results in the longer run.[1]

On the other hand, UNID systems are designed to be more versatile and adaptive, enabling them to recognize a wider range of threats. They use a range of instruments and techniques to locate both recognized and unrecognized attacks. By combining several IDSs, UNID can provide a more full and accurate view of network activity, lowering the possibility of false alarms and missing intrusions.
Instead of lowering the object population below the current levels, these requirements attempted to control its pace of expansion. [1,3,6]

In intrusion detection, machine learning approaches have been widely utilized to identify malicious communications. Sometimes these systems will make mistakes and this can result low accuracy and wrong predictions. 8

II. RELATED WORKS

In this part a brief explanation is given on the related works and the datasets published for NIDs research.

A. NID AND ITS RELATED STUDIES

In this section, the broad concept of NIDS is introduced, and associated research on freely accessible datasets for NIDS R&D is described. IDS, or intrusion detection system is a security tool used to keep an eye on system and network activity in order to identify malicious or unauthorized activity as illustrated in [13, Fig. 1]. IDS come in a variety of forms and can be either software or hardware based. There are three types of intrusion detection systems: signature-based, anomaly-based, and hybrid [1-5]. Signature-based intrusion detection systems (IDS): Also known as knowledge-based intrusion detection systems (IDS), this form of IDS analyses network traffic or system actions to a database of known

attack signatures, patterns, or profiles. The IDS generate an alert if the traffic or behavior matches a signature in the database. As new attacks are identified, the database is regularly updated with new signatures. But Anomaly-based IDS is better at detecting the unknown types of attacks on the network which are not on the database. This is done by creating a baseline of usual activity for a network or system and then searches for deviations from that baseline. Each variation that surpasses a certain threshold is considered a possible attack. Hybrid intrusion detection systems use the advantages of both technologies to improve overall security coverage. [1-2]

It is essential to adopt cutting-edge network intrusion detection systems (NIDs) driven by artificial intelligence (AI) to safeguard corporate networks against cyberattacks, which have recently gotten more varied and complex.
To train AI-powered NIDs, high-quality labelled training datasets are necessary, but creating new training datasets is time-consuming and difficult to find internationally. The Existing system extracts information from the network such as IP Address, Port Number and Time Stamp.[11]

The Data used in the NID is mostly stored as packet data, which is the most common type. It contains network packet headers and payloads, which might contain information about the traffic's source and destination, the protocol employed, and the contents of the packets. NIDS can also analyse metadata, which is data about data. For example, metadata can include information about the size, timestamp, and other characteristics of network packets or other data sources. Flow data is a summary of network traffic that includes information about the source and destination of the traffic, the protocols being used, and the amount of data transferred. NIDS can use flow data to detect anomalies and suspicious patterns of network traffic. [11-12]

The fact that there is a many-to-many link between alerts and the data from API responses is a problem. Therefore, we might think of this connection as one in which numerous alarms from an API answer are tied to one API response. In other words, a communication ow has several alarms connected with it. This indicates that, in order to implement the suggested technique for the alert of NIDSA, the appropriate relationship is first collected via the API. Moreover, one alert is connected to several communication flows. In this instance, one warning may subsequently have an impact on many training labels. Yet, this is a built-in aspect of the system. Certain public datasets predate the alarming amount of attacks used in the recent years, which do not exist on these datasets. Moreover, if the dataset is too small the performance of the NID will also reduce.[1]

B. EVALUATION OF COMMUNICATION FLOWS AND TRIGGER PACKES IN NIDS

Employing cutting-edge network intrusion detection systems(NIDS) driven by artificial intelligence (AI) is essential for safeguarding corporate networks against recent increases in the variety and sophistication of assaults AI-powered NIDS must be trained using high-quality labelled training datasets however these datasets are hard to come by worldwide and creating fresh training datasets is thought to be laborious the authors of the current study look at the viability of a method that combines the advantages of current security appliances to provide labeled training datasets which might be utilized to produce new AI-powered cybersecurity solutions. The authors of the current work begin by detecting communication flows that the installed NIDs view as suspicious, investigating their causes, and uniformly labeling them with the appropriate labels. The packet data in the identified communication flows is then produced by the studies' authors as labelled data along with the necessary alert-type labels.

The authors demonstrate the effectiveness of the labeling scheme by contrasting classification models that were trained using the labeled dataset produced by the authors of the prior study. Additionally, authors of earlier studies provide case studies to examine the effectiveness of a number of regularly used NIDs and to discuss practical solutions to automate the security triage procedure. Labeled datasets for this study were made using open-source NIDs and public datasets to ensure that the results could be repeated. The public is given access to the datasets and software tools for use in research.
The authors used traffic data which T-pot connected with clients in our network and alerts served by NIDSA (Network Intrusion Detection System Evaluation Dataset A) NIDSB (Network Intrusion Detection System Benchmark) NIDSC (Network Intrusion Detection System Corpus)..

C. FLOW BASED INTRUSION DETECTION SYSTEM

Flow-based intrusion detection is now the subject of substantial study. The authors have used a multi-layer perceptron and gravitational search algorithm-based flow-based anomaly detection system. The system has a very high accuracy rate for classifying benign and harmful traffic. The authors suggested a NIDS and obtained a low false alarm rate utilizing a one-class support vector machine for their study. To previous research, the system is trained on a hostile network dataset. It is possible to use intrusion detection algorithms from conventional networks in SDN. Numerous anomaly detection algorithms have also been implemented in the SDN environment to secure the OpenFlow network.

The author's demonstrated how a programmable home network router may offer the best platform for identifying security issues in network by utilizing the programmability of SDN. Rate limitation, utmost entropy detector, and NETAD are four well-known traffic anomaly detection methods that are executed in the SDN environment utilizing OpenFlow compliant switches and a NOX controller. Experiments show that these algorithms are significantly more effective than the ISP at identifying malicious activities in the network, and the anomaly detector can operate at uniform rate without adding any new performance overhead for the traffic on the home network.

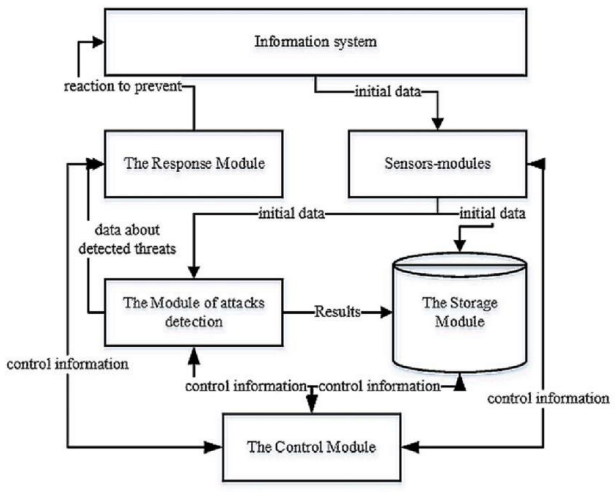

FIGURE 1. Typical Architecture of Network Intrusion Detection System [13]

III. PRELIMNARIES

This section, we will discuss about the preliminaries used in Section IV Proposed Method. We will describe the dataset utilized in this study, followed by the metrics that must be included in the dataset for greater accuracy and reliability.

A. NETWORK TRAFFIC DATA

In Section IV of the Proposed Method we have used KDD Cup Dataset. This dataset was made using DARPA's (Defense Advanced Research Projects Agency) Intrusion Detection and Evaluation systems. Off-line testing was performed on intrusion detection systems utilizing network traffic and audit logs gathered on a simulated network. These data were analyzed in batch mode by the systems, which sought to identify attack sessions in the midst of routine operations.

This dataset contains various attacks namely DoS, DDoS attacks, R2L (Remote-to-User) attack and information gathering attacks such as man-in-the-middle and eavesdropping. These attacks are stores as logs detected by the NIDs with their respective IP Address and port numbers.

B. Measure 1: To determine whether the specified connection sends an alert

It is important to identify from the dataset whether they contain a suspicious or malicious connection within the network. If such a connection is found an alert is produced by the NID and that connection is logged for further analysis. In the case of DoS (Denial-of-Service) and DDoS (Distributed-Denial-of-Service) the NID identifies the source of the attack and blocks traffic from the effected devices. The NIDs then sends an alert that a specified connection is compromised. Traffic filtering and behavioral analysis are used to identify specific attacks by the NIDs. These are the most common approach in detection and prevention of these attacks which are found in the dataset.

IV. PROPOSED METHOD

The predicted decrease in entropy following the split is measured by information gain if the training data is divided based on the values of this feature. Thus, the ability to categories samples more accurately is a property of greater information gain features.

1) DATA PRE-PROCESSING

Pre-processing, the initial step in this stage, is converting data into a format appropriate for deep learning models. Pre-processing decisions must take into account both the type of model being utilized and the data format. When the distances between values indicate some contextual distance, some neural network learning models, such neural networks, often perform at their best. To address these problems, categorical data can be pre-processed using a number of common techniques, such as target encoding or ordinal frequency. The dataset also includes a number of numerical fields in addition to the category ones. Prior to being used in machine learning, numerical data is frequently standardized to guarantee that the inter-feature variation is equal.

Using the above methods, the undesired data from the datasets can be removed as it acts as a critical step in improving the accuracy and reliability of the results obtained from the data. Pre-processing helps to remove noise and irrelevant data, and reduces the impact of outliers, which can improve the accuracy and reliability of analysis results. Data pre-processing can help to organize data in a more structured and consistent way, which makes it easier to analyze and interpret. This can save time and resources in the analysis process. The normalization method is used for a variety of functions, including speeding up the training of classifiers by ensuring that the dataset is consistent and reducing the gap between the data when it exists between huge and small data sets.

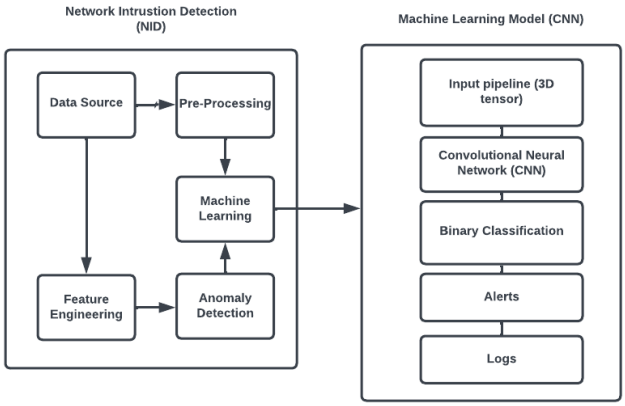

FIGURE 2. CNN model used in Network Intrusion Detection

2) MACHINE LEARNING MODELTRAINING

CNN or Convolutional Neural Network is the main algorithm used for training the dataset to identify if the incoming network traffic is a normal or malicious connection. CNNs are well-suited for analyzing the payload of network packets because they are designed to identify local patterns in the input data. In the context of NID, this involves breaking down the payload of each network packet into smaller segments or "windows" and applying convolutional filters to each segment to identify local patterns. The capacity of CNNs to learn complex feature representations that are challenging to capture using manually constructed rules or signatures is one advantage of their use in NID. This makes it possible for CNNs to identify new or unheard-of attacks that might not be protected by current signatures.

Neural networks are algorithmic approaches that are used to first understand the relationship between two sets of data and then generalize to acquire additional input-output pairs in a sensible manner. In theory, neural networks might be employed in knowledge-based intrusion detection systems to recognize assaults and search them out in the audit stream. Nevertheless, because there is presently no reliable way to interpret what caused the association, the neural network cannot explain the thinking that led to the attack's detection. Neural networks are like a smart system that is capable of finding patterns in large number of different formats unlike statistics. Neural networks learn the regular patterns of that a specific user follows. In this case a UNIX Root user has specific task to be done thus it is easier to predict the regular schedule of this user. Any deviation from these regular tasks can be detected. This helps in recognizing suspicious or malicious activities and to send alerts for these outcomes. Since neural network finds patterns in a large dataset it is much more useful compared to traditional statistical analysis.

V. RESULTS AND EVALUTAION

In this Section we will cover the outcome of the proposed system and evaluate to test the accuracy of the system.

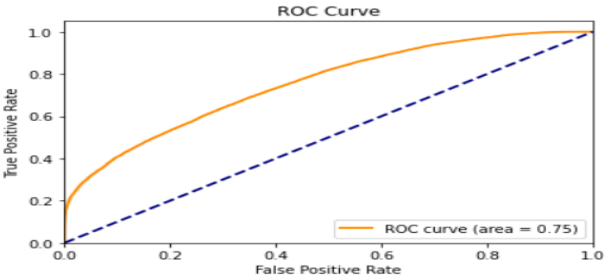

FIGURE 3. The graph shows the ROC Curve

A. TRAINING MODEL ACCURACY

The model is provided with a series of labelled images during the training phase of a CNN, and it learns to extract meaningful features from these images using numerous layers of convolution, pooling, and activation functions. The final layer's output is then sent into a classification layer.

In any machine learning model, there is possibility of false positive in the training model When the model predicts a positive outcome for an input that is actually negative, this is referred to as a false positive. In a binary classification issue, for example, when the aim is to categorize pictures as either containing or not containing a cat, a false positive would arise if the model wrongly predicts that an image without a cat really includes a cat. False positives can arise in a CNN for a variety of reasons, including noisy or unclear data, overfitting, or insufficient training data. These can have serious consequences in particular applications, such as medical diagnosis or fraud detection, where a false positive might result in needless treatments or action Thus, to find the proper accuracy of the training model a separate testing dataset is used. Using this we may acquire a better understanding of the model's behavior and detect possible flaws such as bias, underfitting, or overfitting by assessing it on a different testing dataset. This can assist to increase the model's performance and robustness.

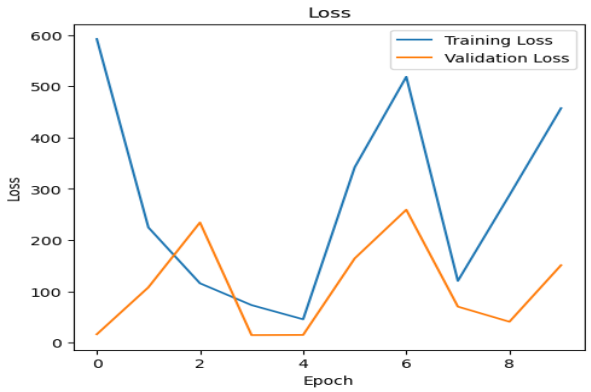

FIGURE 4. The Graph shows the Loss of the Training and validated data

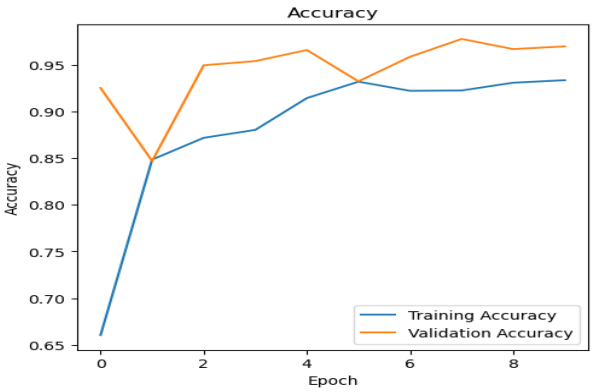

FIGURE 5. The Graph shows the Loss of the Training and validated data

B. EVALUTATION METRIC OF INTEREST

For evaluation it is important to gather a variety of metrics of interest in our trials, including accuracy, precision, recall, and false positive rate. This may derive additional key measures from these, such as the F1 score, the area under the ROC curve, and the location in the precision-recall (PR) space.

A low precision indicates that the detector is misclassifying a substantial proportion of regular traffic as a threat, resulting in too many alarms being sent to network managers or, worse, steps being implemented to deal with these false threats that damage traffic. A low recall, on the other hand, indicates that the detector is unable to distinguish attacks, and the network is not adequately secured. The PR curve is especially useful in imbalanced settings when accuracy alone may provide outcomes that are skewed towards the dominant class. Jupyter Notebook also has the potential to connect with other prominent data science tools such as NumPy, Pandas, and Matplotlib. Users may now do complicated data analysis, generate interactive visualizations, and communicate their findings in a more appealing manner.

With a 0.75 ROC Curve area from Figure 3. The model can determine if a network is malicious or normal for a given piece of data. This demonstrates that the model is functioning and can be used to recognize various networks. These unique types of machine learning approaches can be merged in the future in an IDS to enhance their benefits and overcome their respective shortcomings. Furthermore, developing learning processes with rewards derived from user input or other systems is a viable technique for providing more reliable and resilient intrusion detection systems.

VI. CONCLUSION

The application of artificial intelligence, namely the convolutional neural network (CNN) algorithm, has substantially increased the efficiency and accuracy of network intrusion detection systems. These systems can discover patterns and detect anomalies in network traffic in real time by harnessing the power of machine learning. There are various advantages to applying AI in network intrusion detection systems, including faster threat detection and reaction times, less false positives, and the capacity to identify previously undisclosed or zero-day assaults. Furthermore, AI-based intrusion detection systems can adapt to new and developing threats, making them a significant tool for enterprises and organizations that rely on secure networks to function. Overall, the enlistment of AI in network intrusion detection systems has provided and advantage to network security, delivering more advanced and dependable protection against cyber-attacks.

REFERENCES

[1] Ryosuke Ishibashi, Kohei Miyamoto, Chansu Han , Tao Ban Takeshi Takahashi and Jun'ichi Takeuchi "Generating Labeled Training Datasets Towards Unified Network Intrusion Detection", Systems May 2022

[2] S. Aljawarneh, M. B. Yassein, and M. Aljundi, ``An enhanced J48 classication algorithm for the anomaly intrusion detection systems,"Cluster Comput., vol. 22, no. S5, pp. 1054910565, Sep. 2017

[3] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, ``Toward generating a new intrusion detection dataset and intrusion trafc characterization, '' in Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy, 2018

[4] R.-H. Hwang, M.-C. Peng, V.-L. Nguyen, and Y.-L. Chang, ``An LSTM - based deep learning approach for classifying malicious trafc at thepacket level," Appl. Sci., vol. 9, no. 16, p. 3414, Aug. 2019

[5] H. Hindy, D. Brosset, E. Bayne, A. Seeam, C. Tachtatzis, R. Atkinson,and X. Bellekens, ``A taxonomy of network threats and the

effect of current datasets on intrusion detection systems," IEEE Access, vol. 8, pp. 104650104675, 2020E. E. Reber, R. L. Michell, and C. J. Carter, “Oxygen absorption in the earth’s atmosphere,” Aerospace Corp., Los Angeles, CA, USA, Tech. Rep. TR-0200 (4230-46)-3, Nov. 1988.

[6] Guide to Intrusion Detection and Prevention Systems (IDPS) - Karen Scarfone, Peter Mell, February 2007

[7] Importance of intrusion detection system (IDS)- A. S. Ashoor and S. Gore, January-2011

[8] Deep learning approach for network intrusion detection in soft_x0002_ware defined networking- T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, M. Ghogho, October 2016

[9] Anomaly detection in ad-hoc networks based on deep learning model: A plug and play device -F. Feng, X. Liu, B. Yong, R. Zhou, Q. Zhou, March 2019

[10] A deep learning method with filter-based feature engineering for wireless intrusion detection system -S. M. Kasongo, Y. Sun, March 2019 the Terahertz Wave eBook. ZOmega Terahertz Corp., 2014.

[11] Y. Zhang, X. Chen, L. Jin, X. Wang, and D. Guo, ``Network intrusion detection: Based on deep hierarchical network and originaldata," IEEE Access, vol. 7,2019

[12] 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. Security Defense Appl., Jul. 2009

[13] Sperotto, Anna & Pras, Aiko. (2011). Flow-based intrusion detection. 958-963. 10.1109/INM.2011.5990529.

[14] System Integration and Security of Information Systems Andrii Boikoa, Vira Shendryka,P 2016.

[15] M. Hassan, M. E. Haque, M. E. Tozal, V. Raghavan, and R. Agrawal,``Intrusion detection using payload embeddings," IEEE Access, vol. 10,pp. 40154030, 2022,

[16] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall,D. McClung, D.Weber, S. E.Webster, D.Wyschogrod, R. K. Cunningham,and M. A. Zissman, ``Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation," in Proc. DARPA Inf. Survivability Conf. Expo. (DISCEX), Jan. 2000