International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 12, Issue 09, September 2023)

An Artificial Neural Network Technique for Prediction of Cyber-attack using Intrusion Detection System

Divya Khade[1], Rahul Sahu[2]

[1]*Research Scholar, Department of Computer Science and Engineering, LNCT, Bhopal, India*
[2]*Professor, Department of Computer Science and Engineering, LNCT, Bhopal, India*

[1]divyakhade1407@gmail.com,[2]rahulsahu@lnct.ac.in

***Abstract*-- An intrusion detection system, often known as IDS, is a piece of equipment or a piece of software that monitors a network or collection of devices in order to search for indications of possible intrusion. The frequency of cyber assaults has grown in recent years, and with it, the damage they do to society. The study of cyber security and the avoidance of cyber assaults, such as the use of intrusion detection as a defensive mechanism, is therefore needed. The internet's services are widely used. Services based on computers, the internet, and other forms of technology are all considered part of the cyber world. The cyber world has advanced greatly thanks to new protocols and technologies. Cyber security is a major issue for every service that operates online. Network and host-based intrusion detection systems (NIDS/HIDS) are the backbone of any cyber security infrastructure. The NSL-KDD dataset is often used in algorithm research and verification and is widely employed in both the study and development of intrusion detection systems. In this study, we provide a neural network approach to intrusion detection system threat prediction. The Python Spyder software is used for the simulation.**

***Index Terms* – IOT, Cyber, NIDS, HIDS, Security.**

I. INTRODUCTION

An intrusion detection system (also known as an intrusion prevention system or IPS) is a piece of hardware or a piece of software that is designed to monitor a network or systems for indications of an intrusion or violations of policy. [1] An IDS is also known as an intrusion prevention system or IPS. If an intrusion is found, the system will either alert an administrator or submit the necessary data to a centralised security information and event management (SIEM) database. Both of these actions will take place if the system finds an intrusion. A SIEM system is able to detect harmful behaviour while avoiding false positives because it correlates data from several sources and uses advanced techniques of alert filtering .[2]

Although there are many more types of intrusion detection systems, two of the most common kinds are network intrusion detection systems (NIDS) and host-based intrusion detection systems (HIDS) (HIDS).

Examples of network intrusion detection systems (NIDS) include systems that analyse incoming network traffic, whereas examples of host intrusion detection systems (HIDS) include systems that monitor important operating system files. IDS may also be categorised according to the mechanism that they utilise to detect threats. The most prevalent types are signature-based detection, which identifies hazardous patterns like malware, and anomaly-based detection, which looks for unusual behaviour. Another common kind of detection is the reputation-based variety (recognizing the potential threat according to the reputation scores).

Artificial intelligence (AI) technologies are being used by an increasing number of autonomous control systems to interpret sensor data. This allows the systems to make rapid and accurate judgments on how to carry out control duties depending on the data that has been acquired. In recent years, reinforcement learning (RL) employing deep neural networks has emerged as a viable answer to the challenge of developing artificial intelligence. This is because RL permits learning by interaction with specified restrictions, which was previously impossible. In this investigation, we make use of RL-based control models to find a solution to the issue of fleetingly out-of-date perceptions, which often occurs in environments that are capable of producing digital real effects. Because of this issue, the broad use of RL approaches for decentralised control systems may be hampered. In this paper, we present a reinforcement learning (RL)-based strong control model, or protocol, that makes use of a progressive learning structure. In this structure, a set of low-level strategy variants is first prepared for legacy perceptions, and then their learned information is transferred to an objective environment constrained in ideal information refreshes [5]. In addition, this structure makes use of a progressive learning structure in which a set of low-level strategy variants is first prepared for legacy perceptions.

Artificial intelligence calculations may be useful in certain contexts, but they are less effective when used for interruption identification in digital security.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 12, Issue 09, September 2023)

Digital locators based on AI are vulnerable to hostile attacks like the hassle of beginning examples because of the widespread aversion to their preparation data. Existing controls include exceptions for absurd scenarios, produce poor results even in non-hostile contexts, or may be applied selectively to AI computations that perform insufficiently in terms of digital security [5]. As Internet of Things (IoT) devices and frameworks grow more integral to our interconnected society and environment, they will be increasingly targeted by cybercriminals (including state-supported or affiliated risk entertainers). The breadth and diversity of organisations, the rapidly evolving digital threat landscape, and a host of other factors all contribute to the difficulty of acquiring such devices and systems [6].

II. PROPOSED METHODOLOGY

The proposed methodology is explained using following flow chart-

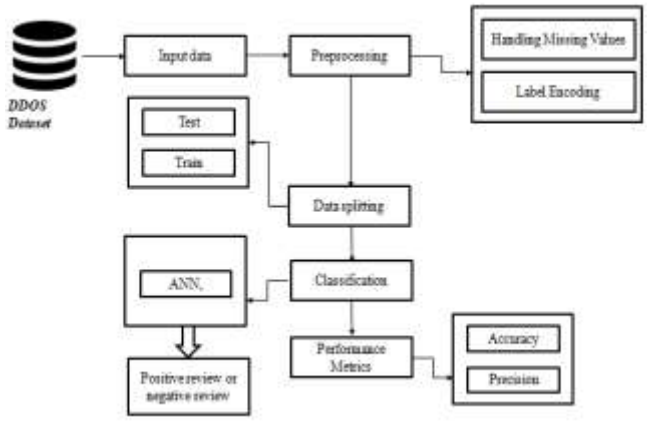

Figure 1: Flow Chart

Steps-

- Firstly, finalize the dataset [13] based on the intrusion detection system, taken from publicly available large dataset repository.
- The data has been preprocessed, and the missing dataset is being sent over right now.
- Now preprocessed data is splitting into the training and testing phase.
- Now artificial neural networks Classification technique is applied
- F-measure, Precision, Accuracy, Recall, and Classification Error are some of the performance criteria you should now evaluate.

These sub-modules form the basis of the proposed research's methodology:

Data Selection and Loading

- The process of picking a dataset and loading it into the Python environment is known as data selections.

Data Pre-processing

- In data pre-processing, the "noise or unwanted data" in a dataset is filtered out.
- Data deficiency correction and categorical data encoding
- The imputer library is used to get rid of any missing or null values in the data.
- Decomposing a Dataset into Test and Training Sets

Splitting Dataset into Train and Test Data

- The term "data splitting" refers to the practise of dividing a dataset into two distinct halves, often for use in a cross-validation setting.
- The data is split in two; one half is used to build a prediction model, and the other half is used to test how well that model performed.

Feature Extraction

Feature extraction is a technique for normalising a set of data's independent variables. Normalization is a procedure that occurs during the pre-processing stage of data processing and goes by another name in the industry.

Classification

ANN- The artificial neurons of an ANN may be thought of as the vertices in a weighted directed graph. Weighted directed edges represent the connection between neuron outputs and inputs. An external source's signal is received by the Artificial Neural Network as a vector representing a pattern and a picture. For each set of n inputs, a mathematical notation x(n) is used to denote the assigned value.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 12, Issue 09, September 2023)

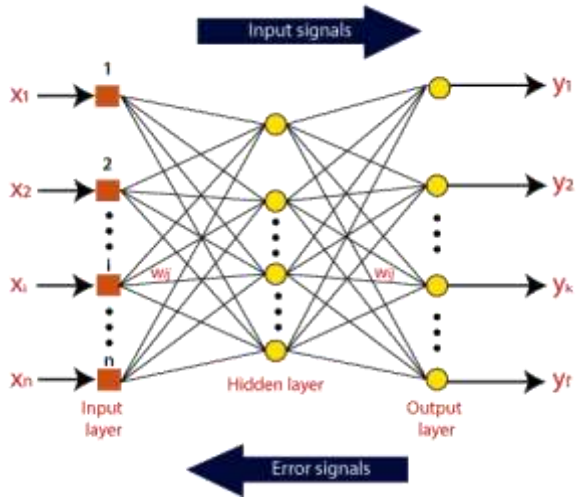

Figure 2: ANN

After that, we multiply each input by its associated weights (these weights are the details utilised by the artificial neural networks to solve a specific problem). These weights often stand for the robustness of the connections between individual neurons in the ANN. The computer device internalises a summary of all the input weights.

With the goal of increasing the system's reaction, bias is introduced if the weighted sum is zero. The input for bias and the value of weight are both 1. Here, the sum of the input weights might be negative or positive infinity. Here, we benchmark a maximum value and run the sum of the weighted inputs through the activation function to constrain the response to acceptable ranges.

Prediction

- This study successfully forecasted the data from the dataset by improving the overall performance of the prediction findings, and it does so by using a technique for predicting intrusion detection.

Algorithm

Input: Intrusion detection Dataset.

Consider the basic information characteristics, such as id with id dur, proto, service, state spkts, dpkts sbytes, dbytes, rate, sttl dttl, sload, etc.

Filtering the null value

Sort the data set according to the characteristics you've chosen.

Output: Best values for F-measure, Precision, Accuracy, Recall, and Classification Error

Step: 1. now dataset is divided into 2 part train and test dataset like train of y and x and test of y and x

2. Extractions of features, features = {} for intrusion count: features [intrusion count] = True

3. Model selection and split

Y train

Y-test

4. Use a classifier based on deep learning's artificial neural network.

5. Confusion matrix with TP, FP, TN, and FN values shown.

6. Determine the percentage of correct answers, standard error, recall, and f-measure.

7. Create a ROC graph.

Evaluation

Accuracy, precision, and recall are the main metrics used to assess a classification model.

- Accuracy is defined as the ratio of true positives to total positives, while recall is defined as the ratio of positives to negatives.
- Accuracy = [TP +TN] / [TP+TN+FP+FN]; F1-Score = 2x (Precision x Recall)/(Precision + Recall)
- Classification Error = 100- Accuracy

Result Generation

The total categorization and prediction will be used to create the final result. Accuracy, error rate, and other similar metrics are used to assess the effectiveness of the suggested method.

III. Simulation And Results

To run the simulation, we use the Python Spyder IDE version 3.7.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 12, Issue 09, September 2023)

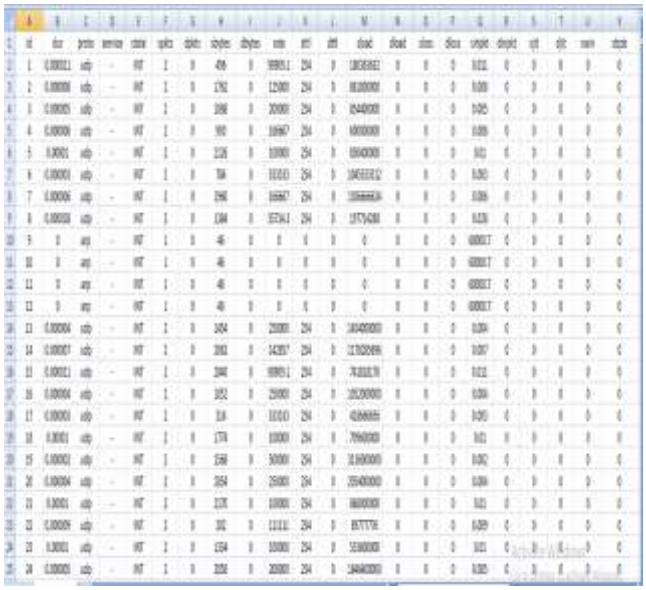

Figure 3: Dataset

The data set is shown in the Python environment (Figure 3). Row and column counts in the dataset might vary widely. Every single column identifies the characteristics by name.

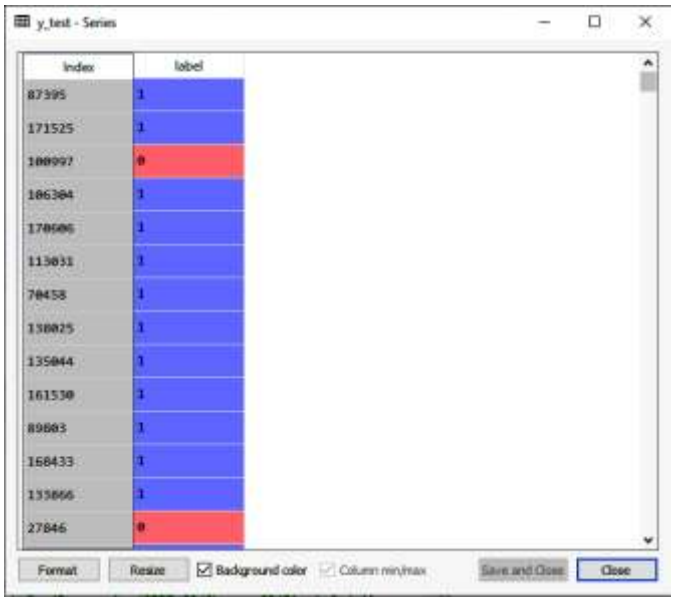

Figure 4: Y test

This dataset's y test is seen in Figure 4. Twenty-thirty percent of the original dataset is used as the train dataset..

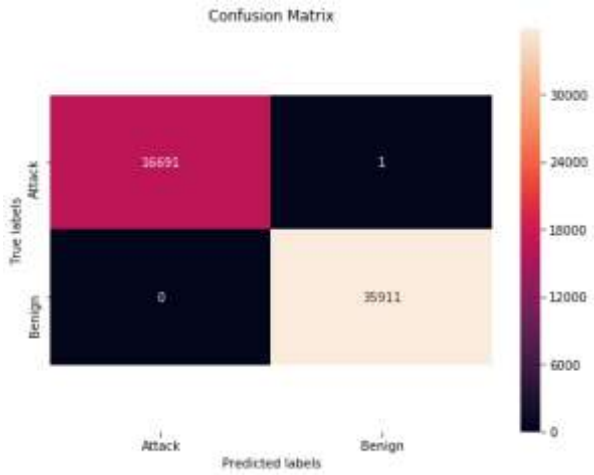

Figure 5: Confusion matrix heat map

Confusion matrices for heat maps generated by the ANN deep learning classification method are shown in Figure 5. It is a matrix of size N by N that measures how well a categorization model does its job.

Table 1:
Simulation Results

Sr. No.	Parameters	Value (%)
1	Precision	99.99
2	Recall	99.99
3	F_Measure	99.99
4	Accuracy	99.99
5	Error Rate	0.01
6	Sensitivity	99.99
7	Specificity	99.99

Table 2:
Result Comparison

Sr No	Parameter	Previous Work [1]	Proposed Work
1	Accuracy	99.94%	99.99%
2	Classification Error	0.06%	0.01%

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 12, Issue 09, September 2023)

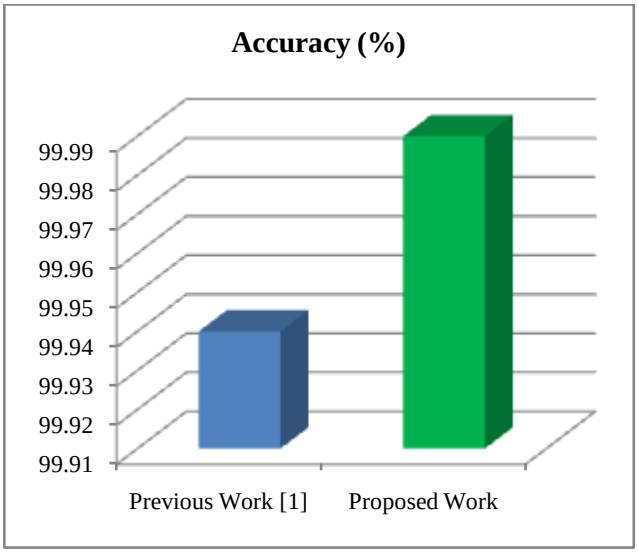

Figure 6: Accuracy Result graph

Figure 6 is presenting the graphical representation of the accuracy. The proposed work achieved better accuracy then existing work

IV. CONCLUSION

Intrusion detection systems, sometimes known as IDSs, are a kind of network security technology that were first created to identify assaults on certain programmes or machines. The cyber world is protected against a wide variety of incursion attempts by the network intrusion system. Artificial intelligence, machine learning, and deep learning are all capable of providing attack prediction strategies. This paper presents an artificial neural network technique for the prediction of cyber-attacks using an intrusion detection system. Simulation is performed using the Python Spyder software. The overall accuracy is achieved at 99.99% by the proposed ANN algorithm with a 0.01% classification error.

REFERENCES

[1] S. Ho, S. A. Jufout, K. Dajani and M. Mozumdar, "A Novel Intrusion Detection Model for Detecting Known and Innovative Cyberattacks Using Convolutional Neural Network," in IEEE Open Journal of the Computer Society, vol. 2, pp. 14-25, 2021, doi: 10.1109/OJCS.2021.3050917.

[2] V. K. Navya, J. Adithi, D. Rudrawal, H. Tailor and N. James, "Intrusion Detection System using Deep Neural Networks (DNN)," 2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), 2021, pp. 1-6, doi: 10.1109/ICAECA52838.2021.9675513.

[3] Y. A. Farrukh, Z. Ahmad, I. Khan and R. M. Elavarasan, "A Sequential Supervised Machine Learning Approach for Cyber Attack Detection in a Smart Grid System," 2021 North American Power Symposium (NAPS), 2021, pp. 1-6, doi: 10.1109/NAPS52732.2021.9654767.

[4] S. Thirimanne, L. Jayawardana, P. Liyanaarachchi and L. Yasakethu, "Comparative Algorithm Analysis for Machine Learning Based Intrusion Detection System," 2021 10th International Conference on Information and Automation for Sustainability (ICIAfS), 2021, pp. 191-196, doi: 10.1109/ICIAfS52090.2021.9605814.

[5] T. T. Nguyen and V. J. Reddi, "Deep Reinforcement Learning for Cyber Security," in IEEE Transactions on Neural Networks and Learning Systems, doi: 10.1109/TNNLS.2021.3121870.

[6] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei and F. Sabrina, "Improving Performance of Autoencoder-Based Network Anomaly Detection on NSL-KDD Dataset," in IEEE Access, vol. 9, pp. 140136-140146, 2021, doi: 10.1109/ACCESS.2021.3116612.

[7] K. Cao, J. Zhu, W. Feng, C. Ma, M. Liu and T. Du, "Network Intrusion Detection based on Dense Dilated Convolutions and Attention Mechanism," 2021 International Wireless Communications and Mobile Computing (IWCMC), 2021, pp. 463-468, doi: 10.1109/IWCMC51323.2021.9498652.

[8] I. Ullah and Q. H. Mahmoud, "Design and Development of a Deep Learning-Based Model for Anomaly Detection in IoT Networks," in IEEE Access, vol. 9, pp. 103906-103926, 2021, doi: 10.1109/ACCESS.2021.3094024.

[9] D. Park, S. Kim, H. Kwon, D. Shin and D. Shin, "Host-Based Intrusion Detection Model Using Siamese Network," in IEEE Access, vol. 9, pp. 76614-76623, 2021, doi: 10.1109/ACCESS.2021.3082160.

[10] I. Siniosoglou, P. Radoglou-Grammatikis, G. Efstathopoulos, P. Fouliras and P. Sarigiannidis, "A Unified Deep Learning Anomaly Detection and Classification Approach for Smart Grid Environments," in IEEE Transactions on Network and Service Management, vol. 18, no. 2, pp. 1137-1151, June 2021, doi: 10.1109/TNSM.2021.3078381.

[11] O. Alkadi, N. Moustafa, B. Turnbull and K. -K. R. Choo, "A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks," in IEEE Internet of Things Journal, vol. 8, no. 12, pp. 9463-9472, 15 June15, 2021, doi: 10.1109/JIOT.2020.2996590.

[12] T. Yu, Z. Liu, Y. Liu, H. Wang and N. Adilov, "A New Feature Selection Method for Intrusion Detection System Dataset – TSDR method," 2020 16th International Conference on Computational Intelligence and Security (CIS), 2020, pp. 362-365, doi: 10.1109/CIS52066.2020.00083.

[13] G. Kadam, S. Parekh, P. Agnihotri, D. Ambawade and P. Bhavathankar, "An Approach to Reduce Uncertainty Problem in Network Intrusion Detection Systems," 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS), 2020, pp. 586-590, doi: 10.1109/ICIIS51140.2020.9342634.

[14] B. Kızıltaş and E. Gül, "Network Anomaly Detection With Convolutional Neural Network Based Auto Encoders," 2020 28th Signal Processing and Communications Applications Conference (SIU), 2020, pp. 1-4, doi: 10.1109/SIU49456.2020.9302202.

[15] https://www.unb.ca/cic/datasets/ids-2017.html