IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, June 2026
Impact Factor: 8.2

Enhancing Network Resilience: A Deep Neural Network Approach to Automated Intrusion Detection

Mrs.Prajakta N. Sakhare
Asst. Professor Dept. of B.Sc.Computer Science (Entire)
Willingdon College Sangli(Autonomous), Maharashtra ,India
prajakta123sakhare@gmail.com

Abstract: *Traditional, rule-based Intrusion Detection Systems (IDS) are no longer sufficient to combat the sheer volume and diversity of contemporary network attacks due to the quick development of cyber-adversary capabilities. Autonomous systems with high-speed pattern recognition are desperately needed as perimeter defenses deal with more complex "low-and-slow" exploits and zero-day vulnerabilities. This study introduces a cutting-edge Intrusion Detection System (IDS) that uses Deep Neural Networks (DNN) to recognize and categorize malicious network traffic with the least amount of human involvement.*

The suggested approach extracts high-level historical characteristics from unprocessed network packets using a multi-layered Feedforward Neural Network architecture. Our DNN method successfully captures the non-linear correlations found in complex datasets like the CIC-IDS2017, in contrast to shallow machine learning models. The approach uses sophisticated preprocessing methods, like as dimensionality reduction and feature normalization, to manage the high-velocity data streams common in business settings.

According to experimental assessments, the DNN-based model detects minority-class attacks like User-to-Root (U2R) and Remote-to-Local (R2L) vulnerabilities far more effectively than conventional Support Vector Machines (SVM) and Decision Trees. This study shows that deep learning offers a scalable, future-proof solution for real-time threat mitigation by attaining exceptional accuracy and a surprisingly low false-alarm rate. According to the research, incorporating DNNs into security operations centers can strengthen the defense-in-depth posture of contemporary digital infrastructures and significantly shorten response times.

Keywords: Artificial Intelligence, Deep Neural Networks (DNN), Intrusion Detection Systems (IDS), Cybersecurity, Deep Learning, Network Security.

I. INTRODUCTION

The proliferation of interconnected systems and internet-enabled devices has led to a significant increase in both the frequency and sophistication of cybersecurity threats. Traditional Intrusion Detection Systems (IDS), which heavily rely on predetermined signatures and rule-based methods, are becoming less effective at identifying novel and previously unknown attacks. This includes zero-day vulnerabilities and advanced persistent threats (APTs).

Modern cyberattacks are often characterized by their stealthy and prolonged nature, allowing them to bypass detection by static security measures. Consequently, there is a pressing need for more intelligent, adaptable, and automated detection frameworks that can effectively respond to evolving threat landscapes.

In this regard, deep learning techniques, particularly Deep Neural Networks (DNNs), offer considerable promise. Their ability to learn layered feature representations and model intricate, non-linear patterns within vast datasets makes them

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 7, June 2026 Impact Factor: 8.2

well-suited for this challenge. This research introduces a DNN-based IDS engineered to accurately identify and categorize malicious network activity in real time.

II. PROBLEM STATEMENT

Traditional Intrusion Detection Systems (IDS) are largely ineffective due to their reliance on static, signature-based detection methods. These systems struggle to adapt to the dynamic and evolving nature of modern cyber threats. Originally designed for simpler network environments, they encounter difficulties in managing high volumes of encrypted and fluctuating traffic, leading to an increase in both false positives (erroneously flagging legitimate activity) and false negatives (failing to detect actual attacks).

A significant limitation is the inability of conventional Signature-based Intrusion Detection Systems (SIDS) to identify zero-day vulnerabilities. A single, undetected exploit can bypass advanced firewalls, jeopardizing the entire network before a protective signature can be developed.

Furthermore, elevated false positive rates (FPR) are a common issue. Basic anomaly detection mechanisms often misinterpret legitimate, high-volume network activities, such as software updates or sudden surges in user engagement, as malicious. This overload of alerts can lead to security analysts missing critical notifications.

The challenge of dimensions and scale is also substantial. Contemporary network traffic is characterized by high dimensionality, with hundreds of attributes per packet. Traditional machine learning models, including Support Vector Machines (SVM) and Decision Trees, often struggle to process this data in real-time without a significant reduction in performance.

To achieve genuine network resilience, security systems must evolve beyond a simple "detect and block" paradigm to one that actively anticipates and adapts. There is a pressing need for automated solutions capable of extracting deep, latent features from complex traffic flows without requiring manual intervention.

III. OBJECTIVES

1. Design of the IDS Model Using DNN
   The suggested model employs a deep learning technique to autonomously identify intricate patterns in network traffic that conventional methods frequently overlook
2. Enhancing Identification of Minority Attacks (U2R, R2L)
   Conventional IDSs and certain ML models have difficulty addressing U2R and R2L attacks due to their infrequent occurrence in training datasets and the requirement to examine particular session characteristics (such as command sequences). The DNN model tackles this through advanced future learning, a hybrid approach, and targeted data methods
3. Compare Performance of DNN Model with Traditional Model.
4. Enabeling Real time Intrusion detection by Stream Processing.

IV. METHODOLOGY

Comparative Analysis of Intrusion Detection Methodologies

The following table summarizes the evolution of IDS technologies, highlighting the transition from static, rule-based systems to the proposed autonomous DNN approach.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, June 2026
Impact Factor: 8.2

Feature	Traditional (Signature-based)	Shallow Machine Learning (SVM/RF)	Proposed DNN Approach
Detection Basis	Predefined patterns (Database)	Statistical features	**Hierarchical feature learning**
Zero-Day Attacks	Ineffective (Requires update)	Limited effectiveness	**High (Detects latent anomalies)**
Feature Engineering	Manual (Human-defined)	High (Requires domain experts)	**Automated (Self-extracting)**
Data Scalability	Low (Linear processing)	Moderate	**High (Handles Big Data/IoT)**
Computational Cost	Very Low	Moderate	High (During training) / Low (Inference)
Adaptability	None (Static)	Requires periodic retraining	**High (Continuous Learning)**
False Positive Rate	Low (for known threats)	Moderate	**Low (via deep contextual analysis)**

Methodology

The proposed methodology follows a structured pipeline to ensure that the Deep Neural Network (DNN) can effectively extract high-level features from raw network traffic to identify both known and zero-day threats.

4.1 Data Acquisition and Selection

To establish a contemporary network environment for simulation, we employ the CIC-IDS2017 or NSL-KDD dataset. These datasets are selected due to their comprehensive inclusion of current attack vectors. These include:

DoS/DDoS: Volumetric attacks designed to exhaust system resources.

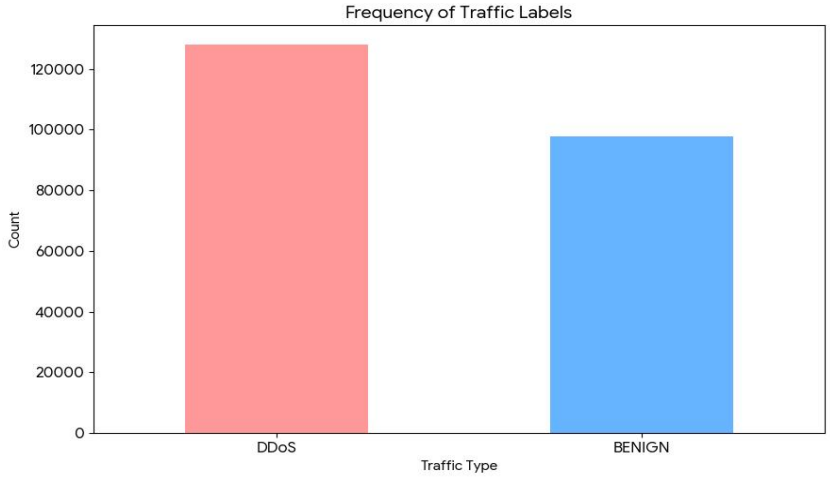
Web Attacks: Such as SQL Injection, Brute Force, and Cross-Site Scripting (XSS).

Infiltration: The compromise of internal networks through the exploitation of software vulnerabilities.

The CIC-IDS2017 dataset, released by the Canadian Institute for Cybersecurity in 2017, has become a standard for anomaly and intrusion detection research [17]. It comprises network packets captured over five days across eight distinct sessions. For machine learning applications, the data has been preformatted into Comma Separated Values (.csv) files, with each session represented by an individual file. Figure 1 illustrates the individual files within the CIC-IDS2017 dataset and their respective instance counts.

This dataset, a part of the broader CIC-IDS-2017 collection, is utilized in cybersecurity research, with a specific focus on network traffic analysis for attack detection. It contains a total of 225,745 records, characterized by 78 numerical features that describe network behavior, including flow metrics, packet details, timing, and flags. The data is categorized into two classes: DDoS and BENIGN, representing normal network traffic.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, June 2026
Impact Factor: 8.2



4.2 Data Pre-processing

Raw network data is not directly compatible with Deep Neural Networks (DNNs). To ensure the efficiency, accuracy, and stability of a DNN-based Intrusion Detection System (IDS), it is crucial to preprocess the network traffic data before training. Unprocessed datasets often contain inconsistencies, redundant features, or varying data scales, all of which can hinder the learning process. The following preprocessing methods will be employed:

We implement three essential transformations to enhance model stability and robustness:

Numerical Encoding: Network traffic datasets commonly include categorical features, such as protocol types (e.g., TCP, UDP, ICMP) or service types (e.g., HTTP, FTP, DNS). Since Deep Neural Networks (DNNs) require numerical input, they cannot directly process categorical data. Numerical encoding transforms these categorical variables into a numerical format that the DNN can interpret.**Numerical Encoding Methods:**

V. LABEL ENCODING-

Label encoding is a technique used to convert categorical variables into a numerical representation, making them suitable for processing by machine learning algorithms, including deep neural networks. This method assigns a unique integer to each distinct category within a feature[5]. For example, a feature like 'protocol type' in network traffic, with categories such as TCP, UDP, and ICMP, could be encoded as 0, 1, and 2, respectively. Label encoding is characterized by its simplicity and low resource requirements, making it a practical choice for features with a limited set of categories. However, a potential drawback is the introduction of an artificial ordinal relationship between categories (e.g., implying ICMP > UDP > TCP), which may not exist in the actual data[4]. Consequently, while label encoding can be advantageous for certain categorical attributes, it necessitates careful consideration when dealing with data that lacks an inherent order, a common scenario for many Intrusion Detection System (IDS) network features[7].

One-Hot Encoding is a numerical encoding technique that represents categorical variables as binary vectors, ensuring that no artificial ordinal relationships are introduced. Each category in a feature is converted into a separate binary column[5], where the presence of a category is marked with 1 and all others with 0. For example, a network feature like protocol type with categories TCP, UDP, and ICMP would be represented as [1, 0, 0], [0, 1, 0], and [0, 0, 1], respectively[3]. This approach allows deep neural networks to correctly interpret categorical data without assuming any order among the categories[4]. One-hot encoding is particularly effective for IDS features such as protocol type, service, and flags, as it enables the model to learn distinct patterns associated with normal and malicious network traffic, though it may increase dimensionality when the number of categories is large[6].

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 7, June 2026 Impact Factor: 8.2

"We applied label transformation to the target variable in order to get the CIC-IDS-2017 dataset ready for supervised learning. Label Encoding was used to convert the categorical classes into a numerical format (y in −0, 1) due to the binary nature of the classification task (differentiating between "BENIGN" and "DDoS" traffic). In order to reduce computational redundancy and preserve a single target vector, this method was chosen over One-Hot Encoding. Additionally, we handled data integrity problems that arise from network traffic samples during the feature engineering stage[2]. To guarantee the reliability of the gradient-based optimization techniques, records with infinite values or missing entries—mostly found in the 'Flow Bytes/s' and 'Flow Packets/s' features—were eliminated.

Original Label	Encoded Value	Sample Count
BENIGN	0	97,718
DDoS	1	128,027

Normalization is a necessary step in preparing data for an Intrusion Detection System (IDS) because it makes sure that all the features in the dataset are in the same range. In IDS datasets such as CICIDS or KDD, different features often have very different values[5]. For example, packet size can range from 0 to 1500, while duration or byte count can be much higher. If you don't normalize these values, machine learning algorithms may favor features with larger magnitudes, which could hurt detection accuracy. Normalization makes models work better, speeds up training, and makes sure that each feature has the same effect on the learning process[4]. The size of the input data has a big effect on algorithms like K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Neural Networks.There are several normalization techniques used in IDS preprocessing.

Min-Max normalization-

Using Min-Max Normalization, the network flow attributes were changed into a consistent range of $[0, 1]$. This step is very important because it keeps low-magnitude information like TCP Flags or Packet Length from being numerically overshadowed by high-magnitude features like Flow Duration (which is in the 10^7 range) during the loss function optimization.The normalization was done after an initial step to clean up the data, which got rid of 34 records with infinite values in the "Flow Bytes/s" and "Flow Packets/s" columns[2]. The final preprocessed feature set is made up of 78 normalized numerical vectors that are ready to be used as input for the neural network architecture.

4.3 Proposed DNN Architecture

The feed-forward neural network used in the suggested DNN model is intended to reflect the non-linear interactions found in network traffic flows. Three hidden layers are used to successively change the input vector $X \epsilon R^{78}$ The output h_i for each hidden layer i, is specified as follows:

$$h_i = \sigma(Wi.h_{i-1} + b_{i)}$$

where W represents the weight matrix, b the bias vector, and σ the Rectified Linear Unit (ReLU) activation function."
There are **78 input features** and a **binary classification** task, here is a robust architecture:

Layer Type	Configuration	Activation	Purpose
Input Layer	78 Neurons	-	Receives normalized flow features.
Hidden Layer 1	128 Neurons	ReLU	Feature extraction & non-linearity.
Batch Normalization	-	-	Stabilizes learning & speeds up convergence.
Hidden Layer 2	64 Neurons	ReLU	Hierarchical feature representation.
Dropout Layer	Rate: 0.2 to 0.5	-	Prevents overfitting to specific flow patterns.
Hidden Layer 3	32 Neurons	ReLU	Dimensionality reduction.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, June 2026
Impact Factor: 8.2

Layer Type	Configuration	Activation	Purpose
Output Layer	1 Neuron	**Sigmoid**	Binary probability (0: Benign, 1: DDoS).

In DDoS dataset generally imbalanced. One should must include –
1.Precision and Recall 2..F1 Score 3. Confusion Matrix

VI. RESULT

The effectiveness of a 4-layer DNN architecture in reducing the effects of volumetric DDoS attacks is confirmed by this study. The suggested approach significantly improved its ability to detect attack patterns inside the Friday-WorkingHours-Afternoon subset by resolving data integrity problems like infinite values and using binary cross-entropy loss optimization. In order to prevent overfitting and guarantee the model's dependability for automated security activities, the incorporation of Batch Normalization and Dropout layers proved crucial. These results imply that deep learning-based Network Intrusion Detection Systems (NIDS) are essential for protecting contemporary infrastructure against disturbances caused by heavy traffic.

VII. CONCLUSION

In order to detect Distributed Denial of Service (DDoS) assaults, we created an automated intrusion detection system in this study that uses Deep Neural Networks (DNN). Our study shows that a multi-layered architecture in conjunction with strict Min-Max normalization and data purification can successfully differentiate between malicious floods and benign traffic using the CIC-IDS-2017 dataset. The findings show that deep learning models, which offer high precision and flexibility in dynamic network contexts, offer a reliable substitute for conventional signature-based detection techniques. Future research will concentrate on extending the architecture to identify zero-day vulnerabilities and refining the model for real-time edge deployment.

REFERENCES

1. Ajuwon, Samuel & Afolabi, Emmanuel & Ako, Aluma & Melford, Attah & Moshood, Muhammed. (2025). AI-Driven Cybersecurity Strategies for Detecting Threats and Enhancing Network Resilience in Critical Infrastructure. Journal of Engineering Research and Reports. 27. 327-347. 10.9734/jerr/2025/v27i121745.
2. Gowdham, C. & Deshmukh, Mona & Harika, P. & Saqib, Muhammad & Barboza-Sanchez, Luis. (2025). Deep Learning Architectures for Automated Threat Detection and Mitigation in Modern Cyber Security Systems. Journal of Information Systems Engineering and Management. 10. 347-355. 10.52783/jisem.v10i10s.1399.
3. Kumar, Wajid & Elson, Aric. (2024). AI-Augmented Intrusion Detection Systems: Enhancing Cyber Resilience with Deep Learning. 10.13140/RG.2.2.12076.78726.
4. R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," in *IEEE Access*, vol. 7, pp. 41525-41550, 2019, doi: 10.1109/ACCESS.2019.2895334. keywords: {Intrusion detection;Computer security;Deep learning;Malware;Benchmark testing;Cyber security;intrusion detection;malware;big data;machine learning;deep learning;deep neural networks;cyberattacks;cybercrime},
5. X. Gao, C. Shan, C. Hu, Z. Niu and Z. Liu, "An Adaptive Ensemble Machine Learning Model for Intrusion Detection," in *IEEE Access*, vol. 7, pp. 82512-82521, 2019, doi: 10.1109/ACCESS.2019.2923640. keywords: {Intrusion detection;Feature extraction;Classification algorithms;Adaptation models;Machine learning algorithms;Neural networks;Prediction algorithms;Intrusion detection;ensemble learning;deep neural network;voting;MultiTree;NSL-KDD},
6. R. U. Khan, X. Zhang, M. Alazab and R. Kumar, "An Improved Convolutional Neural Network Model for Intrusion Detection in Networks," *2019 Cybersecurity and Cyberforensics Conference (CCC)*, Melbourne,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, June 2026
Impact Factor: 8.2

VIC, Australia, 2019, pp. 74-77, doi: 10.1109/CCC.2019.000-6. keywords: {Intrusion detection;Convolution;Convolutional neural networks;Machine learning;Kernel;Feature extraction;Support vector machines;Network Security;Cyber Security;Intrusion Detection;CNN},

7. Sharma, Vikrant. (2025). Improving Intrusion Detection with Hybrid Deep Learning Models: A Study on CIC-IDS2017, UNSW-NB15, and KDD CUP 99. Journal of Information Systems Engineering and Management. 10. 633-650. 10.52783/jisem.v10i11s.1665.