SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 12, No. 5.1, (2026), pp. 942-950
Open Access. Online & Print

DOI: 10.5281/zenodo.12511075

A HYBRID MACHINE LEARNING FRAMEWORK FOR REAL-TIME NETWORK INTRUSION DETECTION IN LARGE-SCALE DATA STREAMS

Nallabariki Praveen Kumar[1]*, R Rajaramesh Merugu[2], V Baby[3], T. Gnana Prakash[4], K. Anusha[5]

[1,4,5] *Assistant Professor, Department of CSE.*
[2] *Professor, Department of Computer Science & Engineering.*
[3] *Associate Professor, Department of CSE.*

Received: 01/12/2025
Accepted: 02/01/2026

Corresponding author: Nallabariki Praveen Kumar (praveenkumar_n@vnrvjiet.in)

ABSTRACT

As the internet is growing very fast so the computer networks are becoming more vulnerable to the cyber-attacks. The traditional security systems are not always effective because they depend on the fixed rules and the common attack patterns so they cannot identify new or the unknown threats that they have never seen before. To solve this problem, the proposed system uses a smart Network Intrusion Detection System (NIDS) which is based on the Artificial Intelligence (AI) that uses a hybrid machine learning approach which makes it more flexible and powerful in identifying the attacks. The system works in two main stages. First, it uses an unsupervised learning method where the model learns only from the normal and the healthy network traffic to understand about the safe behavior and then uses the techniques like Isolation Forest and Autoencoder to identify any unusual or the suspicious activities. Second, it uses a supervised learning method called as XGBoost that takes both the suspicious scores from the first stage and the original network data to make a final and correct decision about whether the activity is an attack or not. The system was tested using a large dataset called CICIDS2017, which contains more than 1.6 million records of network traffic including different types of attacks. The results showed excellent performance with an overall accuracy of 99.87% and a very low false alarm rate of only 0.088% which indicates that the system is highly reliable. When the system is tested with new and unseen attacks the system successfully identified them as anomalies and shows that it can identify zero-day attacks that have never been seen before. In addition, the system includes a real-time alert mechanism that sends the notifications to the administrators or the devices whenever an attack is identified and allows for immediate action to reduce the damage. Overall, this AI-based NIDS provides a solution for protecting digital networks from both known and the unknown cyber threats.

KEYWORDS: Network Intrusion Detection System (NIDS), anomaly detection, hybrid machine learning, Isolation Forest, Autoencoder, XGBoost

1. INTRODUCTION

Cybersecurity has become one of the most important concern in today's digital world. Organizations depend heavily on the computer networks for the process of communication, storing the data, financial transactions, and for using the cloud services. As these network systems become more advanced so there are more chances for the cyber threats to occur. According to the global cybersecurity reports the cyberattacks are increasing in both the number and the difficulty [1], [5] and are creating the serious financial losses and operational problems for the organizations.

The traditional intrusion detection systems mainly use signature-based detection methods [9], [10]. In this approach, the system compares network traffic with the stored patterns of the known attacks. This method works well for identifying the attacks that are already known, but it cannot identify new or the unknown attacks. Also maintaining large databases of the attack patterns increases the system complexity, it requires more computing power, and makes it difficult to handle the system efficiently [4].

Machine learning techniques have become a solution [14] for identifying the network intrusions. These techniques study the patterns in network data and find the unusual behavior that may indicate an attack. Supervised learning methods [4], [16], [17] have shown strong performance in classifying different types of attacks. However, these methods require labelled datasets for training and may not perform well when identifying the zero-day attacks that are never seen before.

Unsupervised learning methods help to solve this problem [7], [18] by identifying the unusual patterns without any need of the labelled data. Techniques like Isolation Forest and the Autoencoders learn the normal network behavior and identify anything that deviates from it. Although these methods are useful, they often produce a high number of false alarms when they are used alone. They also cannot clearly identify the exact type of attack, which reduces their usefulness in the detailed classification tasks. Because of these limitations, the unsupervised methods are usually combined with the supervised techniques [12], [15] to improve accuracy and the overall performance.

To overcome these challenges a real-time hierarchical intrusion detection system is implemented in the SentinelIDS. The system combines both the unsupervised anomaly detection models and a supervised classification model in a two-stage structure. It captures live network traffic, converts it into meaningful features, and analyzes it through the detection system for analysis.

The main contributions of this work include a real-time intrusion detection system that can capture and analyze the live network data and a hybrid machine learning model that combines unsupervised and the supervised techniques, and a method for combining the anomaly scores to improve the classification accuracy. It also includes a scalable detection system that works efficiently with the low delay and provides a complete evaluation using the standard datasets as well as the real-time network data.

1.1. Objectives:

1. To develop a real-time intrusion detection system that can continuously observe the network traffic and can quickly identify any suspicious or the malicious activity.
2. To design a hybrid machine learning model that combines both the unsupervised (anomaly detection) and the supervised (classification) techniques for achieving the better accuracy.
3. To improve the detection of unknown (zero-day) attacks by identifying the unusual patterns in the network behavior without depending only on the known attack data.
4. To reduce the false alarms and to increase the system efficiency by using a two-stage detection process and the optimized feature analysis.

1.2. ACRONYMS

Table 1: Acronyms used in the study

Acronym	Full Form
AI	Artificial Intelligence
NIDS	Network Intrusion Detection System
DDoS	Distributed Denial of Service
DoS	Denial of Service
ML	Machine Learning
DL	Deep Learning
SVM	Support Vector Machine
CNN	Convolutional Neural Network
BiLSTM	Bidirectional Long Short-Term Memory
SDN	Software Defined Network
SIEM	Security Information and Event Management
IF	Isolation Forest
AE	Autoencoder

1.3. Mathematical Symbols

Table 2: Mathematical Symbols

Symbol	Meaning
(X = [x_1, x_2, \dots, x_n])	Feature vector (input features)
(x_i)	Individual feature
(s(x) = 2^{-h(x)/c(n)})	Isolation Forest anomaly score
(h(x))	Path length
(c(n))	Average path length
(L = |x - \hat{x}|^2)	Autoencoder loss (reconstruction error)
(x)	Original input
(\hat{x})	Reconstructed output
(X' = [X, S_{IF}, S_{AE}])	Hybrid feature vector
(S_{IF})	Isolation Forest score
(S_{AE})	Autoencoder score
(\text{Obj} = l(y, \hat{y}) + \Omega(f))	XGBoost objective function
(l(y, \hat{y}))	Loss function
(\Omega(f))	Regularization term
(n)	Number of samples
(d)	Number of features
(t)	Number of trees
(O(t \log n))	Isolation Forest complexity
(O(n \times d))	Autoencoder complexity
(O(n d \log n))	XGBoost complexity

2. LITERATURE REVIEW

Network intrusion detection has been studied a lot in the recent years because the cyber threats are becoming more difficult and dangerous. Many researchers have suggested using the machine learning and the deep learning methods to identify the harmful activities in the network traffic.

In the early days the intrusion detection systems mainly used the signature-based methods [9]. These systems worked by comparing the incoming network traffic with the stored patterns of the known type of attacks. This method was very accurate for identifying the attacks that were already known, but it could not identify the new or the unknown attacks that had not been observed before.

To overcome this problem, the researchers started using the machine learning techniques. The Supervised learning methods such as decision trees, support vector machines, and the random forests [16], [17] became widely used for the network intrusion detection. These methods learn from the labelled datasets and classify the network traffic as either normal or malicious. However, their performance depends a lot on the availability of properly labelled training data, which is not always easy to get.

The Unsupervised learning methods have also been used [7] to identify the anomalies in the network traffic. Techniques like the clustering, Isolation Forest, and the Autoencoders can find the unusual patterns without any need of the labelled data. These methods are especially useful for identifying the zero-day attacks, which are new and the unknown threats. However, when they are used alone the unsupervised techniques can produce a high number of false alarms.

The deep learning methods have also become popular [2], [11] for the intrusion detection tasks. The neural networks and the deep autoencoders can learn the difficult patterns in the network data and have shown better accuracy compared to the traditional machine learning methods in many cases. However, these methods usually require high computational power and the large amounts of training data, which can make them expensive and difficult to use in some of the environments.

The recent research has focused on the hybrid detection systems [3], [13], which combine multiple machine learning techniques to improve the overall performance. These hybrid systems use both the anomaly detection models and the classification algorithms together to reduce the false alarms and to increase the accuracy. For example, some studies combine autoencoders with the gradient boosting classifiers to build the multi-stage detection systems.

A related study on the wearable IoT and machine learning shows how the real-time data analysis can be combined with the intelligent detection models to improve system reliability. Similar ideas can be applied to the cybersecurity, where real-time network data needs to be analyzed quickly and efficiently.

Even though there has been a lot of development in the intrusion detection research but still there are some challenges. Many of the existing systems are tested only on offline datasets and are not implemented in the real-time environments. Also,

some systems have high computational costs and cannot easily handle large and the difficult network systems.

2.1 Research Gap and Novelty

Even after these developments in the machine learning-based intrusion detection systems, there are still some problems in the existing methods. Many systems depend only on the supervised learning methods and they cannot identify new or the unknown (zero-day) attacks easily and whereas the systems that use only unsupervised learning methods often give too many false alarms. Also, most of the research is tested only on the stored (offline) data and does not show how the system works in the real-time situations.

To solve these problems, this work introduces a new hybrid intrusion detection system with the following main features:

- A two-stage system that combines unsupervised methods (Isolation Forest and Autoencoder) with a supervised method (XGBoost) for the better identification.
- A hybrid scoring method that combines anomaly scores with the original data features to improve the accuracy.
- A real-time detection system that can analyze live network traffic with very low delay.
- An adaptive system that can identify both the known attacks and the new (zero-day) attacks effectively.
- A real-time alert system that sends notifications immediately when a threat is identified.

This combination of real-time processing, hybrid learning, and the multi-stage detection makes the proposed system more advanced and effective when compared with the existing intrusion detection systems.

3. METHODOLOGY

3.1. System Architecture

The proposed intrusion detection system is made up of four main parts where each part has a specific role:

1. **Packet Capture Layer –** This part collects the data packets from the network in the real time. It acts like a sensor that listens to all the incoming and outgoing network traffic.
2. **Feature Extraction Layer –** This layer takes the captured data and converts it into the useful information (features) that the system can understand and analyze it.
3. **Anomaly Detection Layer –** In this stage, the system checks the data to find any unusual or suspicious behavior using the methods like Isolation Forest and the Autoencoder.
4. **Classification Layer –** This is the final layer which decides whether the identified activity is normal or an attack by using a model like XGBoost.

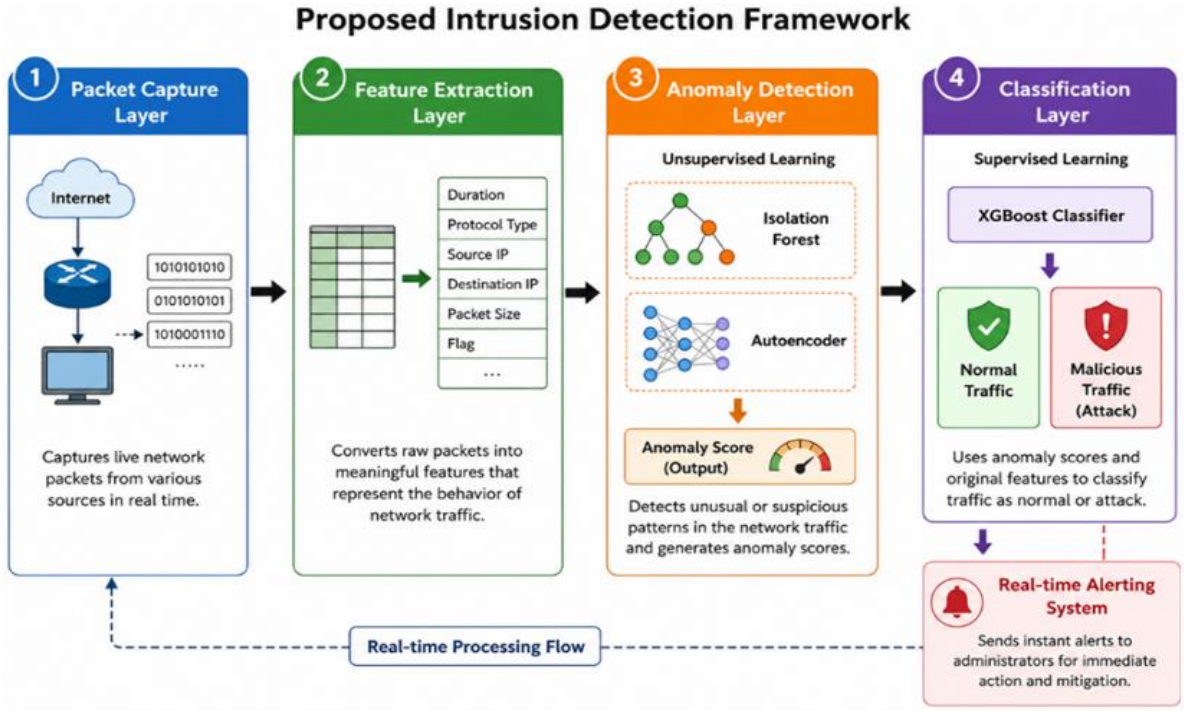

Figure 1: Proposed hybrid intrusion detection framework with real-time processing.

The system collects real-time network traffic using the packet sniffing tools like Scapy, which capture the raw data packets directly from the network. These packets are then analyzed to get the useful

information. The information that we have got is then prepared for the process of training by cleaning the data, normalizing the values, and converting them into a structured format such that models can understand.

Next, this processed data is sent into a two-stage machine learning system. In the first stage, anomaly detection models check the data to find anything unusual or different from the normal network behavior. This helps in identifying both the known attacks and the new or the unseen attacks.

In the second stage, the suspicious data is passed to a classification model like XGBoost. This model identifies the exact type of attack, such as DoS, DDoS, or the brute-force attacks.

This two-stage process helps the system not only to identify the new threats effectively but also to correctly classify the known types of attacks.

3.2 Feature Extraction

Network packets are analyzed to get the useful information which are called as features and they help in understanding the system network behaviour

- **Packet length:** This shows the size of each packet and helps to identify the unusual data patterns.
- **Flow duration:** This measures how long a network connection lasts and helps to identify the connections that are too long or too short.
- **Protocol type:** This tells which communication protocol is used (like TCP or UDP) and gives the context about how the data is being transmitted.
- **Source and destination ports:** These show where the data is coming from and going to and helps to identify the suspicious or the unauthorized access.
- **Packet rate:** This measures how many packets are sent in a certain time that helps to identify the sudden traffic spikes like DDoS attacks.
- **Byte rate:** This shows how much data is transferred over time and helps to identify high or the unusual data usage.
- **Flow statistics:** These are summary values like average (mean) and variation (variance) that describe the overall behavior of the network traffic and helps to differentiate between the normal and the abnormal activity.

All these features together form the input feature vector:

$$X = [x_1, x_2, x_3, \dots, x_n]$$

where each x_i represents one extracted network traffic feature.

3.3. Anomaly Detection Model

3.3.1. Isolation Forest

Isolation Forest is a method used to identify the unusual or the suspicious data points (anomalies) by separating them from normal data. It works by randomly splitting the data into the smaller parts again and again until each data point is isolated. Points that are different from the rest (anomalies) get isolated faster while the normal points take longer.

The anomaly score is calculated using the path length of each data point, which means how many steps it takes to isolate that point in the tree structure, along with the average path length of all the data points.

$$s(x) = 2^{-\frac{h(x)}{c(n)}}$$

Here,

- $h(x)$is the path length (number of steps needed to isolate the data point),
- $c(n)$is the average path length for all data points.

If the score is higher, it means the data point is more likely to be abnormal or an attack, while lower scores indicates the normal behaviour.

3.3.2. Autoencoder Model

Autoencoders are used to identify the unusual or the suspicious data by trying to copy (reconstruct) the input data. They first learn how the normal network traffic looks and then try to recreate it. If the input is normal, the autoencoder can reconstruct it very well. But if the input is abnormal or an attack, it cannot reconstruct it properly, which leads to a higher error.

The difference between the original input and the reconstructed output is called the reconstruction error, and it is calculated using a loss function:

$$L = \| x - \hat{x} \|^2$$

Here,

- $x$is the original input data,
- $\hat{x}$is the reconstructed output.

If the error is large, it means the data is abnormal or suspicious network traffic, while a small error means the data is normal**.**

3.4. Hybrid Feature Fusion

The system takes the anomaly scores from both models (Isolation Forest and Autoencoder) and combines them with the original network features to create a new and the improved input for the classifier.

$$X' = [X, S_{IF}, S_{AE}]$$

Here,

- Xrepresents the original network features,
- $S_{IF}$is the anomaly score from the Isolation Forest,
- $S_{AE}$is the anomaly score from the Autoencoder.

This combined (hybrid) feature set gives more useful information to the model, which helps it to understand better about the difference between the normal and malicious network traffic.

3.5. Classification Model

In the final step, the system uses the XGBoost algorithm to classify the network traffic into different

categories, such as normal traffic or the specific types of attacks.

XGBoost works by trying to make very correct predictions while also keeping the model simple to avoid the overfitting. It does this using an objective function, which has two parts:

$$\text{Obj} = l(y, \hat{y}) + \Omega(f)$$

Here,

- $l(y, \hat{y})$is the loss function, which measures how much the predicted output $\hat{y}$ differs from the actual value y. A smaller value means better prediction.
- $\Omega(f)$is the regularization term, which controls the difficulty of the model so that it does not become too complicated and overfit the data.

In simple terms, XGBoost tries to make accurate predictions while keeping the model simple and efficient.

Algorithm 1: Hybrid Intrusion Detection Framework

Input: Network traffic packets

Output: Predicted attack category

- Capture the network traffic packets in real time using a packet sniffing mechanism.
- Extract flow-level statistical features from the captured packets.
- Preprocess the extracted features through cleaning, normalization, and the encoding.
- Apply the Isolation Forest model to compute an anomaly score for each instance.
- Apply the Autoencoder model to compute reconstruction error for each instance.
- Combine the anomaly scores with the original feature set to form a hybrid feature representation.
- Input the hybrid feature representation into the XGBoost classifier.
- Predict and output the corresponding attack category.

3.6. Complexity Analysis

Let:

- n= number of data samples (records),
- d= number of features (input values),
- t= number of trees used in the model.
- Isolation Forest complexity: $O(t \log n) \rightarrow$ This means the time taken depends up on the number of trees and grows slowly as the dataset size increases which makes it efficient.
- Autoencoder complexity: $O(n \times d) \rightarrow$ This depends on the number of data samples and the features so more data or features increase the training time.
- XGBoost complexity: $O(n\, d \log n) \rightarrow$ This depends on the number of samples, features, and also grows with the size of the data, but remains efficient for the large datasets.

In simple terms, all these methods are developed to work efficiently even with the large amounts of data. Because the system uses a hierarchical (step-by-step) structure, it analyses the data in an organized way which makes it fast enough for the real-time network monitoring and the attack identification.

4. RESULTS AND DISCUSSIONS

The proposed model is evaluated using the CICIDS2017 dataset, which includes a wide range of normal and the attack traffic scenarios. The preprocessing techniques and the evaluation metrics are used to have a comparison of their performance. The system performance is calculated using the important metrics such as Accuracy, Precision, Recall, F1-score, and the ROC-AUC score to have a better understanding of its identification capability and the reliability. The main objective of this evaluation is to analyze how effectively the proposed hybrid method can identify both the known and the unknown attacks while minimizing the false alarms for the real-time applications.

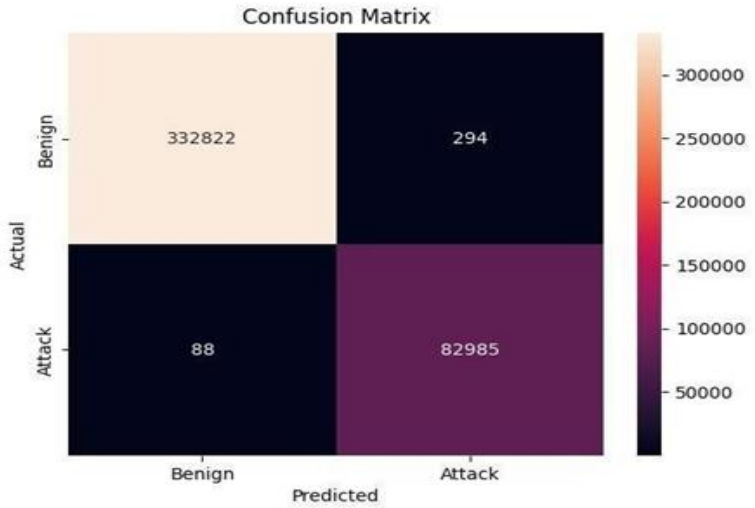

Figure 2: Confusion matrix showing accurate classification of benign and attack traffic with minimal errors.

The confusion matrix shows that the system is performing very well. Most of the values are on the diagonal, which means the system is correctly identifying both the normal (benign) traffic and the attack traffic. There are very few mistakes which indicates that the system has low false positives (normal traffic wrongly marked as attack) and low false negatives (attacks missed as normal).

Table 3: Performance metrics of anomaly detection and attack classification models showing high accuracy and reliability.

Metric	Anomaly Detection	Attack Type Classification
Accuracy	0.9990	0.9980
Precision	0.9966	0.9980
Recall	0.9983	0.9980
F1-Score	0.9975	0.9980
ROC-AUC	1.0000	–
End-to-End Accuracy	0.9987	0.9987

The model shows very good performance with the high precision, recall, accuracy, and the F1-score. This means it can correctly identify the attacks while making very few mistakes such as false alarms or the missed attacks. The high ROC-AUC value also shows that the model is strong and reliable in overall performance.

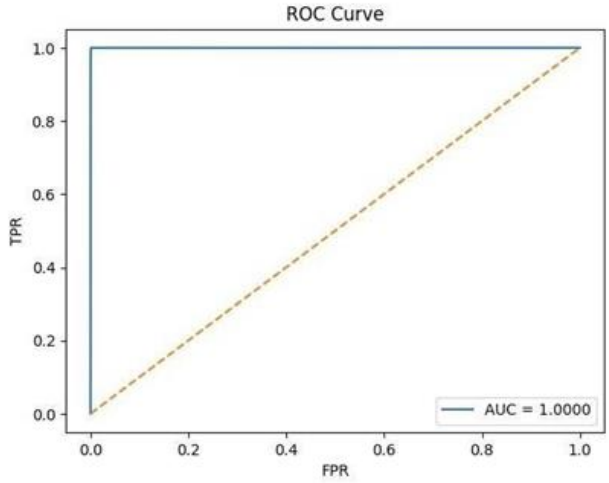

Figure 3: ROC curve showing excellent classification performance with AUC close to 1.

The ROC curve is very close to the top-left corner, which means the model is very good at telling the difference between the normal and the attack traffic. The high ROC-AUC score also shows that the model can clearly separate benign and the malicious data.

The system correctly identifies the normal network traffic and does not give unnecessary alerts. This shows that the system is stable and produces very few false alarms, even when working in the real-time conditions.

Figure 4: Real-time intrusion detection dashboard displaying network traffic monitoring and threat analysis.

The system correctly identifies and classifies known attacks and it can also identify new and the unseen attacks as "unknown". This shows that it can handle both the known and the new threats effectively without making the wrong classifications.

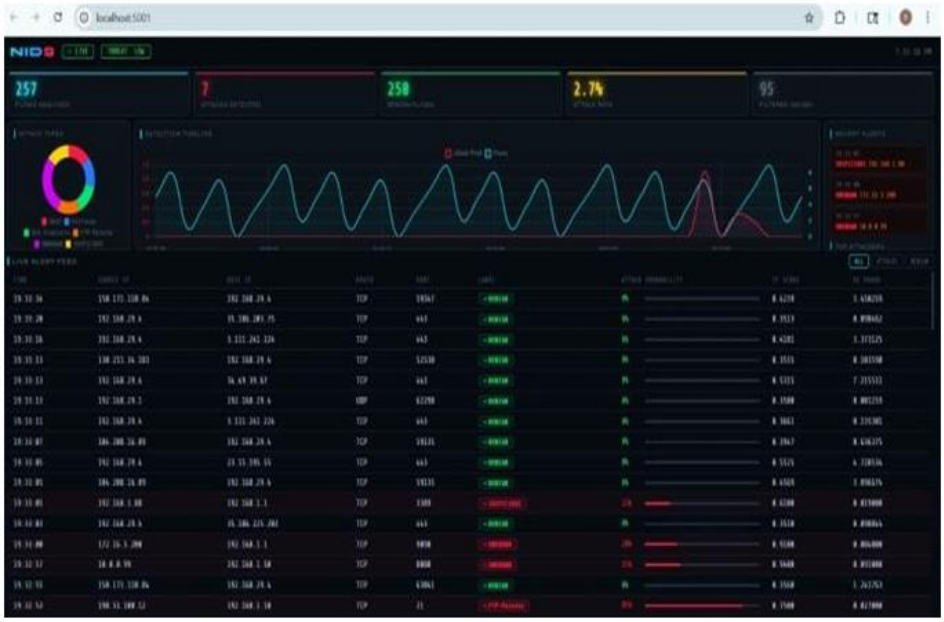

Figure 5: Real-time dashboard showing detection and classification of network attacks with alert visualization.

The system correctly identifies and classifies known attacks, and it can also identify new, unseen attacks as "unknown." This shows that it can handle both known and new threats effectively without making wrong classifications. In addition, the system has an alert feature that sends real-time notifications to the user or administrator whenever any malicious activity is identified and helps them to take the quick action.

4.1 Comparative Results

To check how well the proposed hybrid model works, it was compared with the several other methods, including the single anomaly detection models and the traditional machine learning models. All the models were trained and tested on the same dataset, using the same data preparation steps and evaluation methods, to make sure that the comparison was fair.

Table 4: Comparative performance of different models showing the proposed hybrid model achieving the best results.

Model	Accuracy	Precision	Recall	F1 Score	ROC-AUC
Isolation Forest	0.781125	0.334341	0.082634	0.132516	0.680102
K-Means	0.814107	0.992269	0.081777	0.151101	0
Autoencoder	0.844279	0.965864	0.238718	0.38282	0.935653
Logistic Regression	0.989665	0.984649	0.963945	0.974187	0.99666
IF + AE + Random Forest	0.99864	0.99605	0.997232	0.99664	0.999839
IF + AE + XGBoost (Proposed)	0.998747	0.996422	0.998300	0.99750	0.999924

The proposed hybrid system performs better than all the other methods which shows that it can identify both the known and the unknown attacks very effectively. The results clearly show that the hybrid model gives the best overall performance which makes it highly correct and suitable for intrusion detection.

When compared with the individual models like Isolation Forest and Autoencoder, the hybrid model performs much better which means it can identify more number of attacks without missing them. Traditional methods like the Logistic Regression also perform well, but they are still slightly less effective than the hybrid approach.

By combining the anomaly detection methods with a strong classification model the system is able to learn both the normal and the abnormal patterns in the network traffic. This helps it identify both the known attacks and new or the unseen attacks more effectively and leading to better results across all the performance measures.

5. CONCLUSION

This research tells about a hybrid intrusion detection system that works in real time by combining the unsupervised anomaly detection and the supervised classification methods. The system identfies the live network traffic and analyzes it step by step using a structured machine learning approach. The results show that the system achieves a very high accuracy and produces very few false alarms and can effectively identify both the known attacks and the new (zero-day) attacks. Combining anomaly scores from the different models also helps to improve the classification performance. Because the system works in real time it can handle large amounts of data, and is reliable for the present cybersecurity problems.

6. FUTURE SCOPE

In the future, this work can be improved by using the advanced deep learning approaches such as the transformer-based models to understand better about the difficult patterns in the network traffic over the time. The system can also be tested in large cloud environments to check how well it works on a bigger scale and in the real-world situations. Connecting it with SIEM (Security Information and Event Management) systems can help in centralized monitoring and better analysis of the security threats. In addition, adding the automatic response features can help the system to quickly react when an attack is identified. Finally, by using the federated learning it can allow multiple systems to work together for the intrusion detection while keeping their data private and secure.

REFERENCES

[1] J. Shan and H. Ma, "Optimization of Network Intrusion Detection Model Based on Big Data Analysis," 2024.
[2] Ch. Divya, V. Sulochana, K. V. N. Santhoshi, I. G. Surya, K. Veerendranadh, and J. C. Manikanta, "CNN-BiLSTM: A Hybrid Deep Learning Approach for Network Intrusion Detection in SDN," 2023.
[3] "Design of a Lightweight Network Intrusion Detection System Based on Artificial Intelligence Technology," 2024.
[4] "Machine Learning Approaches for Botnet Detection in Network Traffic," 2024.
[5] Suresh and A. C. Jose, "Adaptive Network Intrusion Detection Using Reinforcement Learning with Proximal Policy Optimization," 2025.
[6] Westphal, S. Hailes, and M. Musolesi, "Feature Selection for Network Intrusion Detection," 2025.
[7] Rookard and A. Khojandi, "Unsupervised Machine Learning for Cybersecurity Anomaly Detection in Traditional and Software-Defined Networking Environments," 2025.
[8] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F.
[9] M. Foozy, "Benchmarking of Machine Learning for Anomaly-Based Intrusion Detection Systems in the CICIDS2017 Dataset," IEEE Access, 2021.
[10] "Machine Learning Approach for Anomaly-Based Intrusion Detection Systems Using Isolation Forest Model and SVM," 2024.
[11] "Deep Learning for Network Anomalies Detection," 2024.
[12] "Real-Time Anomaly Detection in Network Traffic: A Hybrid ML-DL Approach," 2025.
[13] "Using Ensemble Learning, A Cosine Similarity-Based Model for Detecting Security Anomalies in SDN," 2023.
[14] "MLTs-ADCNs: Machine Learning Techniques for Anomaly Detection in Communication Networks," IEEE Access, 2022.
[15] F. Alotaibi and S. Maffeis, "Mateen: Adaptive Ensemble Learning for Network Anomaly Detection," RAID 2024, ACM, 2024.
[16] Z. Liu and Y. Zhou, "Research on SVM Intrusion Detection Algorithm Based on Improved," 2023.
[17] "A Double-Layered Hybrid Approach for Network Intrusion Detection System Using Combined Naive Bayes and SVM," IEEE Access, 2021.
G. Pu, L. Wang, J. Shen, and F. Dong, "A Hybrid Unsupervised Clustering-Based Anomaly Detection Method," Tsinghua Science and Technology, 2021.