# International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 1 (2026)





ijerst.editor@gmail.com

editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 22, No. 1, 2026
© 2026 IJERST. All Rights Reserved
Peer Reviewed, Referred & Indexed Journal

**Research**

# Cyber Attack Detection and Alert System Using Machine Learning

Dr. M.CHAITANYA KISHORE REDDY 1, RAJULAPATI MANOJ 2, SRUNGAVRUKSHM DURGA VARA PRASAD 3, SHAIK MOHAMMAD KAIF 4

#1 Professor ,Dept of Information Technology,,NRI Institute of Technology,Agiripalli

#2#3#4 B.Tech with Specialization of Information Technology in NRI Institute Of Technology,Agiripalli

**Abstract** _ In today's digital age, cyber attacks are one of the most serious risks to individuals, businesses, and governments. Attackers use system vulnerabilities to steal data, destroy infrastructure, or disrupt services. Traditional security solutions frequently fail to detect novel or undiscovered attack patterns in real time. To address this issue, this research presents a Cyber Attack Detection and Alert System based on Machine Learning (ML). The technology examines real-time network traffic and system logs to detect unusual behavior that could suggest a cyberattack. The system uses supervised learning algorithms like Random Forest and Support Vector Machine (SVM) to classify network activity as benign or malicious. In answer to this difficulty, we have built an upgraded approach: an incremental majority voting IDS system, which employs existing tools and methodologies to improve the resilience and adaptability of intrusion detection. Our system aims to improve the accuracy and effectiveness of intrusion detection in real-time scenarios by leveraging the collective decision-making power of multiple machine learning models such as the KNN classifier, Softmax Regressor, and Adaptive Random Forest classifier. This includes reducing false alarm rates. The system was tested using the NSL-KDD and CICIDS2017 datasets, and it obtained 96.43% accuracy and 100% precision for the majority of attack types. It also handles unbalanced data well, making it appropriate for real-world applications. Finally, we improved the project by including a Stacking Classifier to generate even better results.

**Keywords**— **Intrusion Detection System (IDS), Ensemble Machine Learning, Cyber Attack Detection NSL-KDD, CICIDS2017, Real-Time Threat Detection.**

Received: 22-01-2026 Accepted: 26-02-2026 Published: 05-03-2026

## 1. INTRODUCTION

Given the daily rise in sophisticated attacks such as ransomware, advanced persistent threats (APTs), and distributed denial of service (DDoS), intrusion detection is crucial in today's cybersecurity environment. Resilient components are necessary for security systems in order to effectively prevent potential attacks and safeguard network integrity. Intrusion Detection Systems (IDS) are now vital tools for protecting systems against unwanted access

and reducing the risks associated with malicious behavior. The field of intrusion detection has seen a great deal of research employing both machine learning and deep learning techniques to address the network security challenges in order to counter the aforementioned threats. Despite the fact that these studies have demonstrated success in identifying different types of assault, they face two major obstacles. First and foremost, the majority of IDS systems in use today are static and incapable of learning and adapting in real time. This limits their ability to identify recently emerging attacks and necessitates costly retraining processes. Second, these supervised models frequently rely on large, accurately labeled datasets, which can be difficult and expensive to get. Large datasets may also have significant storage requirements. These issues highlight the necessity for resource-efficient and flexible intrusion detection techniques that can effectively handle new threats without requiring a lot of labeling or store space.

## 2. LITERATURE SURVEY

Recent advancements in Network Intrusion Detection Systems (IDS) have increasingly incorporated Artificial Intelligence (AI) and Machine Learning (ML) techniques to address evolving cybersecurity threats. In [1], the authors presented a comprehensive review of AI/ML-based IDS approaches, highlighting the rapid growth of network data and the increasing sophistication of cyberattacks. The study provided a taxonomy of machine learning methods applied to IDS and discussed evaluation metrics such as accuracy, detection rate, and false alarm rate. It also emphasized the persistent challenges of reducing false positives and detecting zero-day attacks. The authors concluded that while ML-based IDS models improve detection performance, further enhancements are required to handle evolving threats and dataset limitations effectively.

Alert correlation and cyber-attack prediction have also gained significant attention in IDS research. In [2], the authors reviewed cyber-attack prediction techniques based on Network Intrusion Detection Systems (NIDS). They highlighted a critical issue in traditional NIDS—namely, the generation of excessive false-positive alerts, with nearly 99% of alerts being low-quality or irrelevant. The study introduced a taxonomy of alert correlation techniques, including similarity-based, statistical-based, knowledge-based, and hybrid approaches. By correlating raw alerts and contextual information, alert correlation systems can predict potential future attacks and provide a high-level view of network security. The paper identified future research directions focused on improving prediction accuracy and reducing alert redundancy.

To overcome the limitations of signature-based systems, hybrid machine learning approaches combining supervised and unsupervised methods have been proposed. In [3], the authors developed a hybrid intrusion detection framework using supervised algorithms such as K-Nearest Neighbors (KNN), Random Forest (RF), and Artificial Neural Networks (ANN) to detect known attacks, while employing the HDBSCAN clustering algorithm to identify unknown or zero-day attacks. Using the CSE-CIC-IDS 2018 dataset with selected features, the supervised models achieved F1-

scores ranging from 0.93 to 0.969 for known attack detection. Meanwhile, the unsupervised clustering approach demonstrated strong performance in identifying previously unseen attacks. This hybrid strategy significantly enhances IDS adaptability in dynamic threat environments.

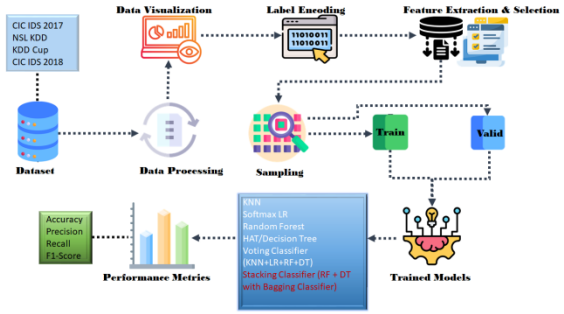
Scalability and real-time detection remain major concerns in IDS research, particularly with the continuous growth of network traffic. In [4], the authors conducted an empirical comparison between batch learning and data streaming algorithms for network intrusion detection. Batch learning models such as J48 and PART were compared with streaming algorithms including Hoeffding Tree (HT) and OzaBagAdwin (OBA). Experimental results showed that streaming algorithms outperformed batch models in binary classification tasks, achieving accuracy levels as high as 99.67%. The study recommended the adoption of data streaming approaches for real-time intrusion detection, as they address scalability issues and reduce retraining overhead.

Incremental learning has further been explored to improve IDS efficiency and adaptability. In [5], the authors proposed AB-HT, an ensemble incremental learning algorithm that combines Adaptive Boosting (AdaBoost) with Hoeffding Tree (HT). Unlike traditional batch models that require retraining with large datasets, AB-HT can detect new intrusion patterns without retraining on historical data. Experimental comparisons demonstrated that AB-HT achieved shorter training times and improved performance, with an average 18% higher F1-score than standalone HT and HATT models. These findings highlight the effectiveness of incremental ensemble learning for dynamic and large-scale intrusion detection environments

## 3. METHODOLOGY

Our suggested system employs a Majority Voting Classifier-based Intrusion Detection System (IDS) that is constantly adaptable to new cyber threats. Rather than relying on a single algorithm, the system integrates the decisions of several machine learning models to make more accurate predictions. The Voting Classifier enables the system to learn from streaming data and respond to changing attack patterns in real time. This increases the system's reliability and eliminates the need for retraining. It also performs well with huge and imbalanced datasets such as CICIDS2017, resulting in improved identification of both common and uncommon assaults. Overall, our technique delivers a scalable, cost-effective, and highly accurate IDS appropriate for modern, dynamic network environments..

**ii) System Architecture:**




**.Fig.1. Proposed Architecture**

The proposed intrusion detection framework begins with benchmark datasets such as CIC IDS 2017, NSL-KDD, KDD Cup, and CIC IDS 2018, which contain labeled records of normal and malicious network traffic. The raw data undergoes preprocessing, including cleaning, normalization, and formatting,

followed by data visualization to analyze traffic patterns and class distributions. Categorical labels are converted into numerical form using label encoding, and feature extraction and selection techniques are applied to identify the most relevant attributes, thereby improving efficiency and reducing dimensionality. To address class imbalance, appropriate sampling techniques are employed before splitting the dataset into training and validation sets. Multiple machine learning models—including K-Nearest Neighbors (KNN), Softmax Logistic Regression, Random Forest, Hoeffding Tree/Decision Tree, Voting Classifier (KNN + LR + RF + DT), and a Stacking Classifier (RF + DT with Bagging)—are trained to detect cyber attacks. Ensemble methods enhance detection accuracy and reduce false positives by leveraging the strengths of individual classifiers. Finally, the system's performance is evaluated using standard metrics such as Accuracy, Precision, Recall, and F1-Score, ensuring reliable and scalable real-time intrusion detection**iii)**

**I. MODULES:**

1. **Data loading:** using this module we are going to import the dataset.
2. **Data Processing:** Data processing involves removing duplicate data and performing drop cleaning to ensure data quality. This step prepares the dataset for further analysis by eliminating inconsistencies and inaccuracies, setting a solid foundation for effective modeling and decision-making.
3. **Data Visualization:** Data visualization entails creating graphical representations of the dataset to uncover patterns, trends, and anomalies. By utilizing various charts and plots, this process aids in comprehensively understanding the data distribution and relationships, facilitating informed decisions in subsequent analysis.
4. **Label Encoding:** Label encoding is the process of converting categorical string data into numerical integers. This transformation allows the machine learning models to interpret the data effectively, enabling them to utilize categorical features as input for training and predictions.
5. **Feature Extraction:** Feature extraction involves selecting relevant features from the dataset, specifically distinguishing between input features (X) and the target variable (y). This step is crucial for enhancing model performance by ensuring that only significant information is utilized during training.
6. **Feature Selection:** Feature selection using mutual information assesses the relevance of features in relation to the target variable. By identifying and retaining only the most informative features, this process improves model accuracy and reduces overfitting, ensuring that the model learns effectively.
7. **Sampling:** Sampling involves preparing the dataset to address class imbalance through various techniques. This includes utilizing original data, applying random under-sampling to reduce majority class instances, random over-sampling to increase minority class instances, and combining both under and over-sampling methods for balance.
8. **Splitting data into train & valid:** using this module data will be divided into train & valid

9. **Model generation:** Model building - KNN - Softmax LR - Random Forest - HAT/Decision Tree - Voting Classifier (KNN+LR+RF+DT) - Stacking Classifier (RF + DT with Bagging Classifier). Performance evaluation metrics for each algorithm is calculated.
10. **User signup & login:** Using this module will get registration and login
11. **User input:** Using this module will give input for prediction
12. **Prediction:** final predicted displayed

**ii) ALGORITHMS:**

K Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is a straightforward yet effective classification technique that finds the 'k' nearest training samples to a given data point and assigns the most common class to them. In the framework of the incremental majority voting Intrusion Detection System, KNN is used to classify incoming network traffic based on past data, making it especially suitable for detecting various forms of intrusions by using the spatial distribution of data points in feature space.

Softmax Logistic Regression (LR)

Softmax Logistic Regression is a variant of logistic regression designed for multi-class classification issues. It calculates the probabilities for each class and assigns the class with the highest likelihood to a data point. Softmax LR is used in the IDS project to classify network data via feature extraction, offering a probabilistic framework for evaluating the likelihood of various forms of intrusions and improving the system's ability to distinguish between normal and malicious activity.

Random Forest

Random Forest is an ensemble learning method that creates many decision trees during training and then combines their predictions to improve accuracy and resilience. Random Forest is used in incremental majority voting IDS to model network traffic and detect intrusions, taking advantage of its ability to handle high-dimensional data and reduce overfitting. Its ensemble nature enables for improved generalization across different assault circumstances, which considerably improves the system's overall detection performance.

HAT / Decision Tree

The HAT (Highly Adaptive Tree) method is a type of decision tree whose structure changes depending on the complexity and distribution of the input. HAT is used in IDS to develop interpretable models capable of accurately classifying network traffic. By constantly responding to changing input patterns, HAT increases the model's flexibility in identifying breaches while offering clear decision paths that cybersecurity experts can easily understand and confirm.

Voting Classifier: (KNN+LR+RF+DT)

The Voting Classifier improves classification accuracy by combining predictions from numerous algorithms, including KNN, Softmax LR, Random Forest, and HAT. In the IDS project, this ensemble technique combines the strengths of multiple models, allowing the system to benefit from a variety of data classification perspectives. The Voting Classifier's use of majority voting strengthens its robustness against many forms of invasions, resulting in a more reliable detection system.

Stacking Classifier (RF+DT with Bagging Classifier)

The Stacking Classifier combines Random Forest and HAT predictions, which are further improved by a Bagging Classifier for even greater stability and accuracy. The incremental majority voting IDS uses each model's strengths to generate a meta-model that learns from the base classifiers' predictions. This approach effectively captures complex patterns in network traffic data, resulting in increased intrusion detection capabilities and overall system performance.

## 4. EXPERIMENTAL RESULTS

Experimental findings show that the Attention-based CNN model classifies earthquake precursor signals better than classical machine learning and CNN. Logistic Regression, KNN, SVM, Decision Tree, Random Forest, XGBoost, CNN, and extended Attention-CNN were tested on 429 seismic waves. Attention-CNN performed best with 99.81% accuracy, compared to 97.98% for the regular CNN model. This increase shows how well the attention mechanism captures crucial temporal relationships and emphasizes relevant signal segments for ramping and non-ramping pattern categorization.

Additionally, accuracy, recall, F1-score, and ROC analysis corroborate the model's robustness and dependability. The attention mechanism prioritizes relevant precursor patterns and reduces noise to improve feature representation. The Flask-based web solution also allows users to submit seismic data and get quick categorization results. The extended system is ideal for sophisticated seismic early warning systems because to its accuracy, generalization, and real-time performance.

**Accuracy:** The ability of a test to differentiate between healthy and sick instances is a measure of its accuracy. Find the proportion of analysed cases with true positives and true negatives to get a sense of the test's accuracy. Based on the calculations:

Accuracy = TP + TN /(TP + TN + FP + FN)

$$Accuracy = \frac{(TN + TP)}{T}$$

**Precision:** The accuracy rate of a classification or number of positive cases is known as precision. Accuracy is determined by applying the following formula:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\Pr e\,cision = \frac{TP}{(TP + FP)}$$

**Recall:** The recall of a model is a measure of its capacity to identify all occurrences of a relevant machine learning class. A model's ability to detect class instances is shown by the ratio of correctly predicted positive observations to the total number of positives.

$$Recall = \frac{TP}{(FN + TP)}$$

**F1-Score:** A high F1 score indicates that a machine learning model is accurate. Improving model accuracy by integrating recall and precision. How often a model gets a dataset prediction right is measured by the accuracy statistic..

$$F1 = 2 \cdot \frac{(Recall \cdot \Pr e\,cision)}{(Recall + \Pr e\,cision)}$$

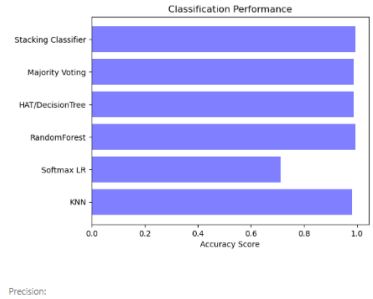



**Fig2 Accuracy Comparision**

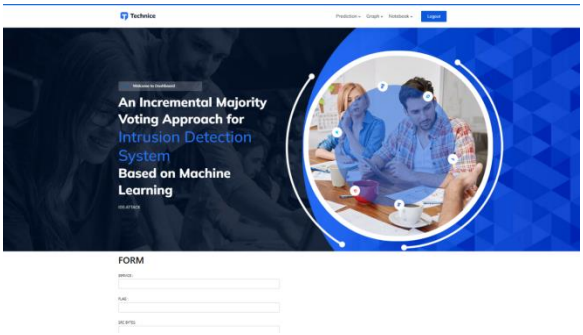



**Fig 3: Input data**




**Fig 4:Predict output.**

## 5. CONCLUSION

Finally, our study overcomes the shortcomings of standard static and single classifier machine learning techniques in detecting intrusion threats by presenting an Incremental Ensemble Learning (IEL) method. By focusing on concept drift in continuous data streams and using the CICIDS2017 dataset, we were able to obtain high accuracy and remarkable prediction accuracy in attack detection. The voting-based IDS model outperformed the HAT, ARF, KNN, and Softmax Regressor algorithms, as well as the ensemble model built with these algorithms. This model outperforms individual classifiers and has a noteworthy detection rate. Notably, our top-performing model attained an overall accuracy of approximately 96.43%, with a fantastic prediction accuracy score of 100% in the Web Attack and Bot attack categories. Despite significantly decreased performance for bot assaults, we have laid a solid platform for future expansions and improvements. Importantly, our model obtained a remarkable low False Alarm Rate (FAR) of only 2%, demonstrating its effectiveness in distinguishing between legal and possibly harmful activity. In future study, we will focus on improving the model's robustness in recognizing new attacks. By continuing our efforts, we hope to push the boundaries of IDS and achieve even greater detection scores while retaining a low FAR, thus contributing to the evolution of intrusion detection technology...

## REFERENCES

[1] P. Vanin, T. Newe, L. L. Dhirani, E. O'Connell, D. O'Shea, B. Lee, and M. Rao, ''A study of network intrusion detection systems using artificial intelligence/machine learning,'' Appl. Sci., vol. 12, no. 22, p. 11752, Nov. 2022, doi: 10.3390/app122211752.

[2] H. Albasheer, M. Md Siraj, A. Mubarakali, O. Elsier Tayfour, S. Salih, M. Hamdan, S. Khan, A. Zainal, and S. Kamarudeen, ''Cyber-attack prediction based on network intrusion detection systems for alert correlation techniques: A survey,'' Sensors, vol. 22, no. 4, p. 1494, Feb. 2022.

[3] L. Shahbandayeva, U. Mammadzada, I. Manafova, S. Jafarli, and A. Z. Adamov,

‘‘Network intrusion detection using supervised and unsupervised machine learning,’’ in Proc. IEEE 16th Int. Conf. Appl. Inf. Commun. Technol. (AICT), Oct. 2022, pp. 1–7.

[4] K. S. Adewole, T. T. Salau-Ibrahim, A. L. Imoize, I. D. Oladipo, M. AbdulRaheem, J. B. Awotunde, A. O. Balogun, R. M. Isiaka, and T. O. Aro, ‘‘Empirical analysis of data streaming and batch learning models for network intrusion detection,’’ Electronics, vol. 11, no. 19, p. 3109, Sep. 2022.

[5] M. Data and M. Aritsugi, ‘‘AB-HT: An ensemble incremental learning algorithm for network intrusion detection systems,’’ in Proc. Int. Conf. Data Sci. Appl. (ICoDSA), Jul. 2022, pp. 47–52.

[6] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, ‘‘Network intrusion detection system: A systematic study of machine learning and deep learning approaches,’’ Trans. Emerg. Telecommun. Technol., vol. 32, no. 1, p. e4150, Jan. 2021.

[7] H. Liu and B. Lang, ‘‘Machine learning and deep learning methods for intrusion detection systems: A survey,’’ Appl. Sci., vol. 9, no. 20, p. 4396, Oct. 2019, doi: 10.3390/app9204396..

[8] M. Mijwil, O. J. Unogwu, Y. Filali, I. Bala, and H. Al-Shahwani, ‘‘Exploring the top five evolving threats in cybersecurity: An in-depth overview,’’ Mesopotamian J. Cyber Secur., vol. 2023, pp. 57–63, Mar. 2023.

[9] S. Gamage and J. Samarabandu, ‘‘Deep learning methods in network intrusion detection: A survey and an objective comparison,’’ J. Netw. Comput. Appl., vol. 169, Nov. 2020, Art. no. 102767. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804520302411

[10] T. Fadziso, U. Thaduri, S. Dekkati, V. Ballamudi, and H. Desamsetti, ‘‘Evolution of the cyber security threat: An overview of the scale of cyber threat,’’ Digitalization Sustainability Rev., vol. 3, no. 1, pp. 1–12, 2023.

[11] D. Bhosale and R. Ade, ‘‘Intrusion detection using incremental learning from streaming imbalanced data,’’ Int. J. Manag. Public Sector Inf. Commun. Technol., vol. 6, no. 1, pp. 9–20, Mar. 2015.

[12] M. R. Mohamed, A. A. Nasr, I. F. Tarrad, and M. Z. Abdulmageed, ‘‘Exploiting incremental classifiers for the training of an adaptive intrusion detection model,’’ Int. J. Netw. Secur., vol. 21, no. 2, pp. 275–289, 2019.

[13] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, ‘‘Learning under concept drift: A review,’’ IEEE Trans. Knowl. Data Eng., vol. 31, no. 12, pp. 2346–2363, Dec. 2018.

[14] S. Wares, J. Isaacs, and E. Elyan, ‘‘Data stream mining: Methods and challenges for handling concept drift,’’ Social Netw. Appl. Sci., vol. 1, no. 11, pp. 1–19, Nov. 2019.

[15] X. Yuan, R. Wang, Y. Zhuang, K. Zhu, and J. Hao, ‘‘A concept drift based ensemble incremental learning approach for intrusion detection,’’ in Proc. IEEE Int. Conf. Internet Things (iThings), IEEE Green Comput. Commun. (GreenCom), IEEE Cyber, Phys. Social Comput. (CPSCom), IEEE Smart Data (SmartData), Mar. 2018, pp. 350–357.