International Journal of Science, Technology and Management (IJSTM) ISSN (online): 2321-774X
Volume 13, Issue 1, 2025

Network Intrusion Detection Using Machine Learning Techniques

Muskan Student, B.Tech. CSE 8th Semester Mr. Vikash, Assistant Professor
BRCM College Of Engineering & Technology Bahal, Bhiwani,Haryana

Abstract

With the rapid growth of internet-connected systems, network security has become a critical concern for organizations and individuals alike. Traditional signature-based Intrusion Detection Systems (IDS) struggle to identify novel and evolving cyber-attacks. Machine Learning (ML) techniques offer a promising alternative by enabling systems to learn patterns from network traffic and detect both known and previously unseen intrusions. This paper presents a comprehensive study of Network Intrusion Detection Systems built using machine learning techniques. It discusses the types of intrusion detection systems, commonly used datasets such as NSL-KDD, KDD Cup 99, and CICIDS2017, and evaluates various ML algorithms including Decision Trees, Random Forest, Support Vector Machines, Naive Bayes, K-Nearest Neighbors, Artificial Neural Networks, and Deep Learning models such as CNN and LSTM. The paper further discusses the general methodology for building an ML-based IDS, evaluation metrics, challenges such as class imbalance and high dimensionality, and future research directions including hybrid and ensemble models.

Keywords: *Network Security, Intrusion Detection System (IDS), Machine Learning, Deep Learning, NSL-KDD, Anomaly Detection, Cyber-Attacks*

1. Introduction

The exponential growth of internet usage and networked devices has significantly increased the attack surface for cyber threats. Organizations face a wide variety of attacks, including Denial of Service (DoS), Probe, Remote to Local (R2L), User to Root (U2R), malware, and increasingly sophisticated Advanced Persistent Threats (APTs). An Intrusion Detection System (IDS) is a security mechanism designed to monitor network or system activities for malicious actions or policy violations and to alert administrators when such activities are detected.

Traditional IDS approaches are broadly classified into two categories: signature-based (misuse) detection and anomaly-based detection. Signature-based systems compare incoming traffic against a database of known attack signatures; while accurate for known threats, they fail to detect new or zero-day attacks. Anomaly-based systems build a model of normal behaviour and flag deviations from it, which allows detection of unknown attacks

but often at the cost of a higher false positive rate.

Machine Learning has emerged as a powerful tool to overcome the limitations of traditional IDS. By learning from large volumes of labelled or unlabelled network traffic data, ML-based IDS can automatically extract patterns that distinguish normal traffic from malicious traffic, adapt to new attack types, and reduce the manual effort required to maintain signature databases. This paper reviews the application of ML techniques to network intrusion detection, compares their performance characteristics, and outlines a general framework for building an effective ML-based IDS.

2. Objectives of the Study

- To study the fundamental concepts of network intrusion detection and its classification.
- To review widely used benchmark datasets for IDS research.
- To analyse the working principles of major machine learning algorithms applied to intrusion detection.
- To compare algorithms based on accuracy, false positive rate, and computational efficiency.
- To identify open challenges and propose future research directions.

3. Related Work

Numerous studies have explored the use of machine learning for intrusion detection over the past two decades. Early work focused on applying classical classifiers such as Decision Trees and Naive Bayes to the KDD Cup 99 dataset, achieving reasonable accuracy but suffering from redundant records and class imbalance issues inherent in the dataset. The NSL-KDD dataset was later introduced as an improved version that removes duplicate records and provides a more balanced distribution of attack classes, making it a more reliable benchmark.

Subsequent research shifted toward ensemble methods such as Random Forest and boosting algorithms, which combine multiple weak learners to improve detection accuracy and reduce overfitting. More recent studies have applied deep learning architectures, including Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and autoencoders, which can automatically learn hierarchical feature representations from raw or minimally processed traffic data, often outperforming classical ML approaches on modern datasets such as CICIDS2017 and UNSW-NB15. Hybrid approaches that combine feature selection techniques with ensemble or deep learning classifiers have also been shown to improve both detection accuracy and computational efficiency.

4. Classification of Intrusion Detection Systems

4.1 Based on Detection Method

Signature-based (Misuse) Detection: Detects attacks by matching traffic patterns against a

database of known attack signatures. High accuracy for known attacks but ineffective against zero-day threats.

Anomaly-based Detection: Builds a baseline model of normal behaviour and flags significant deviations as potential intrusions. Capable of detecting unknown attacks but prone to higher false positive rates.

Hybrid Detection: Combines signature-based and anomaly-based approaches to leverage the strengths of both methods.

4.2 Based on Deployment

Network-based IDS (NIDS): Monitors traffic across the entire network by analysing packets at strategic points such as routers or gateways.

Host-based IDS (HIDS): Monitors activities on an individual host, including system calls, file changes, and application logs.

5. Benchmark Datasets

KDD Cup 99: One of the earliest and most widely used datasets, containing about 4.9 million connection records labelled as normal or one of 22 attack types. It suffers from redundant records and imbalanced class distribution.

NSL-KDD: An improved version of KDD Cup 99 that removes duplicate records from the training and test sets, resulting in a more balanced and reliable dataset for evaluating classifier performance.

CICIDS2017: A modern dataset developed by the Canadian Institute for Cybersecurity that includes realistic benign traffic along with common contemporary attacks such as DDoS, Brute Force, Infiltration, and Botnet activity.

UNSW-NB15: Contains a mix of real modern normal activities and synthetically generated contemporary attack behaviours, widely used for evaluating deep learning-based IDS.

6. General Methodology for ML-Based IDS

A typical machine learning-based intrusion detection pipeline consists of the following stages:

- Data Collection: Gathering network traffic data from live networks or benchmark datasets.
- Data Preprocessing: Handling missing values, encoding categorical features, and normalizing numerical features.
- Feature Selection/Extraction: Reducing dimensionality using techniques such as Information Gain, Principal Component Analysis (PCA), or correlation-based selection to remove irrelevant or redundant features.
- Model Training: Training classifiers such as Decision Trees, SVM, Random Forest, or deep neural networks on the processed dataset.
- Model Evaluation: Assessing performance using metrics such as accuracy, precision, recall, F1-score, and false positive rate.
- Deployment: Integrating the trained model into a real-time or near-real-time monitoring pipeline.

7. Machine Learning Techniques for Intrusion Detection

This section briefly discusses the working principle of major algorithms applied to intrusion detection.

7.1 Decision Tree: A tree-structured classifier that splits data based on feature values to arrive at a classification decision. It is fast and easy to interpret but can overfit on complex datasets.

7.2 Random Forest: An ensemble of decision trees trained on random subsets of data and features; predictions are aggregated through majority voting, improving robustness and reducing overfitting compared to a single tree.

7.3 Support Vector Machine (SVM): Constructs an optimal hyperplane that separates classes with maximum margin. Performs well in high-dimensional feature spaces, making it suitable for traffic data with many features.

7.4 Naive Bayes: A probabilistic classifier based on Bayes' theorem, assuming feature independence. It is computationally efficient, making it suitable for real-time detection despite the independence assumption rarely holding in practice.

7.5 K-Nearest Neighbors (KNN): Classifies a data point based on the majority class among its k nearest neighbours in feature space. Simple to implement but computationally expensive for large datasets.

7.6 Artificial Neural Networks (ANN): Composed of interconnected layers of neurons that learn non-linear relationships between input features and output classes through backpropagation.

7.7 Deep Learning (CNN/LSTM/Autoencoders): Deep architectures can automatically learn hierarchical feature representations from raw traffic data. CNNs are effective at capturing spatial patterns, LSTMs capture temporal/sequential dependencies in traffic flows, and autoencoders are used for unsupervised anomaly detection by learning to reconstruct normal traffic and flagging high reconstruction error as anomalous.

8. Comparative Analysis of Algorithms

The table below summarizes typical performance characteristics reported in the literature for various ML algorithms evaluated on benchmark IDS datasets such as NSL-KDD and CICIDS2017. Exact figures vary across studies depending on the dataset, feature selection method, and experimental setup, and should therefore be treated as indicative rather than absolute.

Algorithm	Detection Accuracy	False Positive Rate	Key Strength
Decision Tree (C4.5)	~95-97%	Moderate	Fast, interpretable
Random Forest	~98-99%	Low	Robust to noise, handles high dimensionality

Support Vector Machine	~96-98%	Low-Moderate	Effective in high-dimensional space
Naive Bayes	~90-93%	High	Very fast, low memory
K-Nearest Neighbors	~93-96%	Moderate	Simple, no training phase
Artificial Neural Network	~97-99%	Low	Captures non-linear patterns
Deep Learning (CNN/LSTM)	~98-99.5%	Very Low	Automatic feature learning, best for large datasets

9. Evaluation Metrics

Accuracy: The proportion of correctly classified instances (both normal and attack) out of the total instances.

Precision: The proportion of correctly identified attack instances out of all instances predicted as attacks.

Recall (Detection Rate): The proportion of actual attack instances that are correctly identified.

F1-Score: The harmonic mean of precision and recall, useful when class distribution is imbalanced.

False Positive Rate (FPR): The proportion of normal instances incorrectly classified as attacks; a critical metric since a high FPR reduces the practical usability of an IDS.

10. Challenges and Open Issues

Class Imbalance: Attack classes such as U2R and R2L are typically underrepresented in datasets, making it difficult for classifiers to learn their patterns effectively.

High Dimensionality: Network traffic data often contains a large number of features, increasing computational cost and the risk of overfitting.

Zero-Day Attacks: Detecting previously unseen attack patterns remains a significant challenge for supervised learning approaches trained only on known attack types.

Real-Time Processing: High-speed networks generate large volumes of traffic, requiring detection models that are both accurate and computationally efficient enough for real-time deployment.

Adversarial Attacks: Attackers may craft traffic specifically designed to evade ML-based detection models, necessitating research into adversarially robust IDS.

Dataset Limitations: Many benchmark datasets are outdated and do not reflect current attack patterns, motivating the need for continuously updated, realistic datasets.

11. Future Research Directions

Hybrid and Ensemble Models: Combining multiple classifiers or combining signature-based and anomaly-based approaches to improve overall detection accuracy and reduce false positives.

Federated Learning: Enabling collaborative model training across multiple organizations without sharing raw traffic data, preserving privacy while improving model generalization.

Explainable AI (XAI): Incorporating interpretability techniques so that security analysts can understand and trust the reasoning behind ML-based alerts.

Online/Incremental Learning: Developing models that can continuously adapt to evolving traffic patterns and new attack types without requiring complete retraining.

Edge and IoT-focused IDS: Designing lightweight ML models suitable for deployment on resource-constrained IoT and edge devices.

12. Conclusion

Machine learning techniques have significantly advanced the field of network intrusion detection by enabling systems to learn from data and adapt to evolving threats, addressing key limitations of traditional signature-based approaches. This paper reviewed the classification of IDS, commonly used benchmark datasets, and the working principles of major ML algorithms, along with a comparative analysis of their performance characteristics. While classical algorithms such as Random Forest and SVM continue to offer strong, interpretable performance, deep learning approaches show great promise for handling large-scale, high-dimensional, and evolving traffic data. Challenges such as class imbalance, high dimensionality, real-time processing requirements, and adversarial robustness remain active areas of research. Future work combining hybrid models, explainable AI, and federated or online learning approaches is expected to further improve the accuracy, efficiency, and trustworthiness of ML-based intrusion detection systems.

References

[1] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. IEEE Symposium on Computational Intelligence for Security and Defense Applications.

[2] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. International Conference on Information Systems Security and Privacy (ICISSP).

[3] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. Military Communications and Information Systems Conference (MilCIS).

[4] Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153-1176.

[5] Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep learning approach for intelligent intrusion detection system. IEEE Access, 7, 41525-41550.

[6] Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Transactions on Emerging Telecommunications Technologies, 32(1).

[7] Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. IEEE Symposium on Security and Privacy.