Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

A COMPARATIVE MACHINE LEARNING APPROACH TO NETWORK FLOW CLASSIFICATION FOR TELECOMMUNICATION CYBERSECURITY

Sabina Norbekova Tolib qizi
Tashkent University of Information Technologies
Named After Muhammad al-Khwarizmi Tashkent, Uzbekistan
Email: sabinanorbekova1211@gmail.com

Abstract

Telecommunication networks generate large volumes of heterogeneous traffic across mobile, broadband, enterprise, cloud, and service-provider environments. As these networks expand, detecting malicious network flows becomes essential for service continuity, infrastructure protection, and cybersecurity operations. This study presents a comparative machine learning approach to binary network flow classification for telecommunication cybersecurity using the UNSW-NB15 dataset. Unlike web request-based detection studies, the research focuses on flow-level characteristics such as duration, protocol, connection state, packet and byte counts, traffic rate, load, time-to-live values, and connection-related statistics. Five supervised models were evaluated: Logistic Regression, Decision Tree, Random Forest, Linear Support Vector Machine, and Gradient Boosting. The workflow included data loading, train-test partitioning, categorical feature encoding, numerical scaling, model training, evaluation, confusion matrix analysis, and feature importance interpretation. Random Forest achieved the best overall performance with accuracy of 0.8921, precision of 0.8510, recall of 0.9746, and F1-score of 0.9086. Gradient Boosting achieved the highest recall of 0.9853, but its lower precision indicates a stronger false alarm tendency. Feature importance analysis showed that connection state, time-to-live information, traffic load, flow rate, duration, byte count, and packet statistics were among the most influential predictors. The findings suggest that tree-based models can

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

support intrusion detection in telecommunication cybersecurity when flow-level monitoring is required at scale.

Keywords: Telecommunication cybersecurity; network flow classification; intrusion detection; machine learning; UNSW-NB15; Random Forest; network traffic analysis; flow-based detection

1. Introduction

Telecommunication networks are critical infrastructure for modern digital communication. They support mobile services, internet access, cloud connectivity, enterprise communication, online platforms, and machine-to-machine data exchange. The International Telecommunication Union estimated that 5.5 billion people were online in 2024, representing 68% of the world population [15]. This level of connectivity increases both the social value of telecommunications and the security responsibility of network operators.

Cybersecurity monitoring in telecommunication environments is difficult because network traffic is high-volume, heterogeneous, and continuously changing. Network operators must distinguish normal behavior from attacks without relying only on manual inspection or application-layer payload content. This becomes more difficult when encryption, protocol diversity, tunneling, and rapidly changing services limit direct inspection. Attacks such as scanning, exploit attempts, denial-of-service activity, suspicious connections, and abnormal traffic bursts may first appear as changes in flow-level behavior.

Network flow classification assigns traffic records to categories such as normal or attack traffic based on statistical and behavioral characteristics of communication. Flow-level features include duration, protocol type, connection state, packet and byte counts, source and destination load, traffic rate, time-to-live values, and connection counters. These indicators are relevant to telecommunication cybersecurity because they can be extracted across protocols and services and integrated into intrusion detection or security operations workflows.

Many machine learning studies evaluate cybersecurity models on benchmark datasets, but some focus on application-layer indicators such as URLs, HTTP

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

request fields, or payload-based patterns. Such approaches are useful for web application security, but they do not fully represent the flow-based monitoring requirements of telecommunication networks. In addition, model performance is sometimes discussed mainly through accuracy, although precision, recall, F1-score, and confusion matrix analysis are more informative for operational cybersecurity.

The aim of this study is to compare supervised machine learning models for binary network flow classification and evaluate their suitability for telecommunication cybersecurity. The main research questions are: how effectively can supervised models classify normal and attack flows; which model provides the best balance between precision, recall, and F1-score; which flow-level features contribute most strongly to classification; and what do the results imply for network monitoring and security operations?

The contribution of this study is a comparative experimental evaluation of five supervised machine learning models on the UNSW-NB15 dataset, interpreted specifically from a telecommunication cybersecurity perspective. The study focuses on flow-level features, evaluates model performance beyond accuracy, and uses Random Forest feature importance to identify the most influential network traffic indicators.

2. Related Work

Intrusion detection has long been an important component of network security. Traditional intrusion detection systems monitor network activities and generate alerts when traffic patterns match known signatures or abnormal conditions. Signature-based detection is effective against known attacks, but it is less reliable against new or modified threats. Anomaly-based detection can identify deviations from normal behavior, but it may generate high false alarm rates when legitimate network behavior changes [3], [5].

Machine learning has become a widely studied approach in network intrusion detection because it can learn patterns from labelled traffic data. Logistic Regression and Linear Support Vector Machine are useful baselines because they are computationally efficient, while tree-based models such as Decision Tree, Random Forest, and Gradient Boosting can capture nonlinear interactions in

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

structured data [4], [6], [8]. In network traffic classification, nonlinear behavior may arise from combinations of protocol, state, duration, packet count, byte count, traffic rate, load, and connection statistics.

Telecommunication cybersecurity requires scalable analysis of large traffic volumes. Machine learning can support this requirement by automating traffic classification, identifying abnormal flow patterns, and prioritizing suspicious traffic for further analysis. However, practical security operations require a balance between missed attacks and false alarms. High recall helps reduce missed attacks, while high precision reduces unnecessary alerts and analyst workload.

Benchmark datasets are important for reproducible intrusion detection experiments. Older datasets such as KDD Cup 99 and NSL-KDD are widely known but have been criticized for outdated traffic patterns and limited representation of modern threats [11], [12]. More recent datasets such as UNSW-NB15 and CICIDS2017 provide structured network flow records and attack labels that are useful for evaluating machine learning-based intrusion detection [1], [2], [10]. This study uses UNSW-NB15 because it includes flow-level and connection-level features and supports both binary and multiclass intrusion detection research.

3. Materials and Methods

3.1 Dataset Description

This study uses the UNSW-NB15 dataset for binary network flow classification. The dataset contains structured traffic records representing normal and attack behavior and includes predefined training and testing partitions. The available partitions were arranged so that 175,341 records were used for training and 82,332 records were used for testing. Each record contained 45 columns, including an identifier, traffic features, an attack category, and a binary class label. The binary label was used as the target variable, where normal traffic was represented as class 0 and attack traffic as class 1.

The dataset includes categorical features such as proto, service, and state, representing protocol type, network service, and connection state. Numerical features include duration, packet counts, byte counts, traffic rate, source and destination load, time-to-live values, packet inter-arrival statistics, TCP timing

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

values, and connection-related counters. These variables represent flow-level behavior and are therefore suitable for telecommunication cybersecurity analysis.

Table 1. Distribution of normal and attack traffic in the UNSW-NB15 dataset

Dataset partition	Normal traffic	Attack traffic	Total records
Training set	56,000	119,341	175,341
Testing set	37,000	45,332	82,332

3.2 Data Preprocessing and Feature Preparation

Data preprocessing was performed in Python. The id column was removed because it does not contain predictive network behavior information. The attack_cat column was excluded because the experiment was designed as binary classification rather than multiclass attack category classification. The label column was used as the target variable. Missing values were checked before model training. Categorical variables were transformed using one-hot encoding, while numerical variables were standardized using feature scaling.

This study did not create manual application-layer features such as URL length, suspicious keywords, or HTTP method indicators. Instead, it used the original flow-level features available in UNSW-NB15. This decision keeps the experiment aligned with network monitoring rather than web application request analysis. After preprocessing, categorical variables were expanded into binary indicator variables, and numerical features were scaled so that linear models could be trained more effectively.

3.3 Machine Learning Models and Metrics

Five supervised models were evaluated: Logistic Regression, Decision Tree, Random Forest, Linear Support Vector Machine, and Gradient Boosting. Logistic Regression and Linear SVM were used as linear baseline models. Decision Tree was included because it can learn rule-based nonlinear patterns. Random Forest was used as an ensemble tree-based model that combines multiple decision trees and supports feature importance analysis [6]. Gradient Boosting was included as a sequential ensemble method that builds trees to correct previous errors [8]. All models were implemented using scikit-learn [9].

The models were evaluated using accuracy, precision, recall, F1-score, and confusion matrix analysis. Accuracy measures overall correctness, precision

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

measures how many predicted attacks were actually attacks, recall measures how many actual attacks were detected, and F1-score balances precision and recall. In cybersecurity, recall is important because missed attacks can be harmful, while precision is important because too many false positives can overload analysts.

Table 2. Experimental workflow

Stage	Description
Dataset loading	Load UNSW-NB15 CSV files
Partition verification	Use 175,341 records for training and 82,332 records for testing
Feature preparation	Remove id, attack_cat, and target label from feature matrix
Preprocessing	Apply one-hot encoding and standard scaling
Model training	Train five supervised machine learning models
Evaluation	Calculate accuracy, precision, recall, F1-score, and confusion matrix
Interpretation	Analyze model comparison and Random Forest feature importance

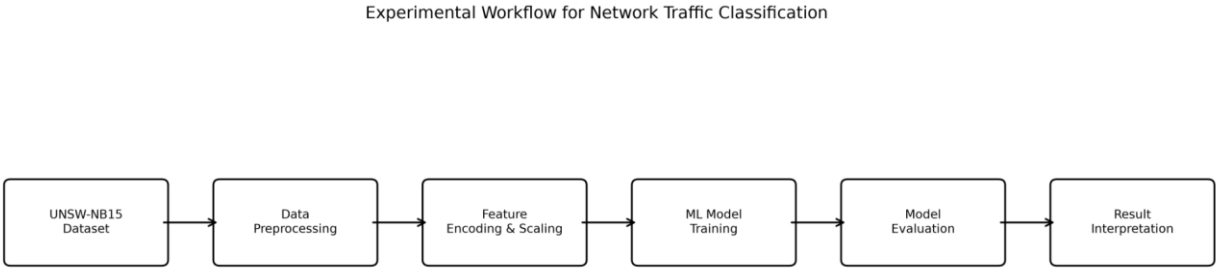

Figure 1. Experimental workflow for network traffic classification.

4. Experimental Results

4.1 Dataset Distribution and Model Comparison

The experiment used the available training and testing partitions of the UNSW-NB15 dataset. The training set contained 56,000 normal records and 119,341 attack records, while the testing set contained 37,000 normal records and 45,332 attack records. This distribution reflects a moderately imbalanced cybersecurity classification problem, so evaluation should not rely on accuracy alone.

Five supervised models were trained and evaluated using the same preprocessing pipeline. Table 3 presents the performance comparison. Random Forest achieved the best overall result with accuracy of 0.8921, precision of 0.8510, recall of 0.9746, and F1-score of 0.9086. Gradient Boosting achieved the highest recall of 0.9853, but its lower precision indicates a stronger false alarm tendency. Logistic

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

Regression and Linear SVM obtained reasonable recall values, but their lower F1-scores show that linear decision boundaries were less effective than tree-based models for this task.

Table 3. Performance comparison of machine learning models on the UNSW-NB15 dataset

Model	Accuracy	Precision	Recall	F1-score
Random Forest	0.8921	0.8510	0.9746	0.9086
Decision Tree	0.8609	0.8162	0.9646	0.8842
Gradient Boosting	0.8563	0.8000	0.9853	0.8831
Logistic Regression	0.8353	0.8020	0.9306	0.8615
Linear SVM	0.8300	0.7939	0.9336	0.8581

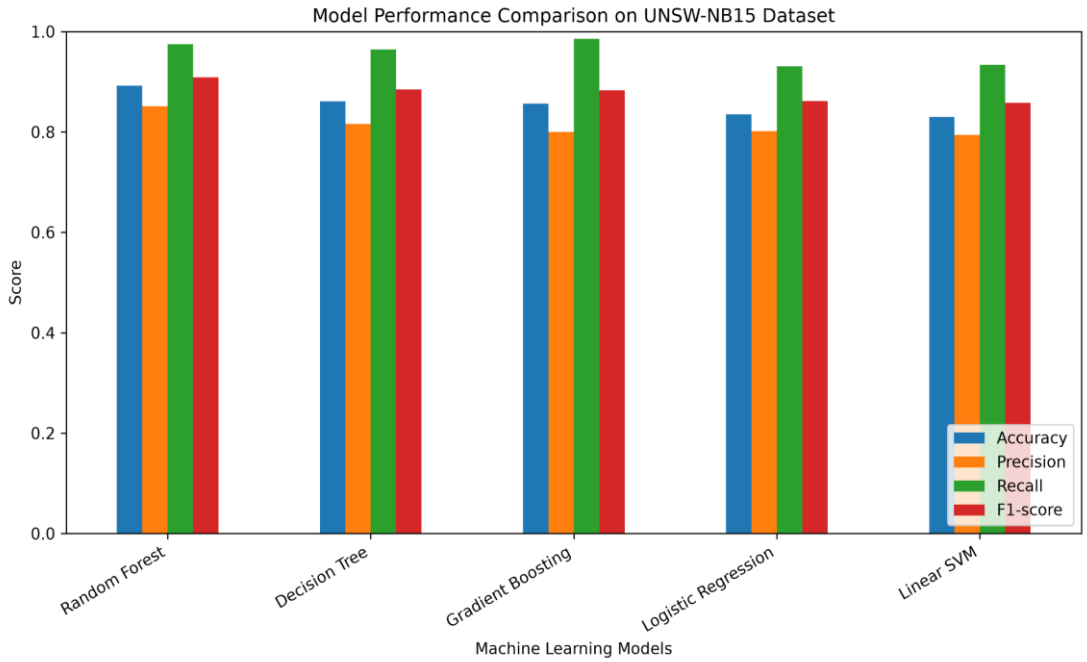

Figure 2. Model performance comparison on the UNSW-NB15 dataset.

4.2 Confusion Matrix Analysis

Since Random Forest achieved the highest F1-score and the strongest overall balance among the evaluated models, its confusion matrix was analyzed in detail. The model correctly classified 29,266 normal records as normal and 44,181 attack records as attacks. It incorrectly classified 7,734 normal records as attacks and 1,151 attack records as normal.

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

Table 4. Confusion matrix of the Random Forest model

Actual / Predicted	Predicted Normal	Predicted Attack
Actual Normal	29,266	7,734
Actual Attack	1,151	44,181

From a cybersecurity perspective, the 1,151 missed attack records are critical because they represent malicious flows that were not detected. However, compared with the total number of attack records in the testing set, the model detected a large majority of malicious traffic. The 7,734 false positives also require attention because excessive alerts can increase the workload of security analysts. Therefore, the model should be used as part of a broader intrusion detection and monitoring system rather than as a standalone decision mechanism.

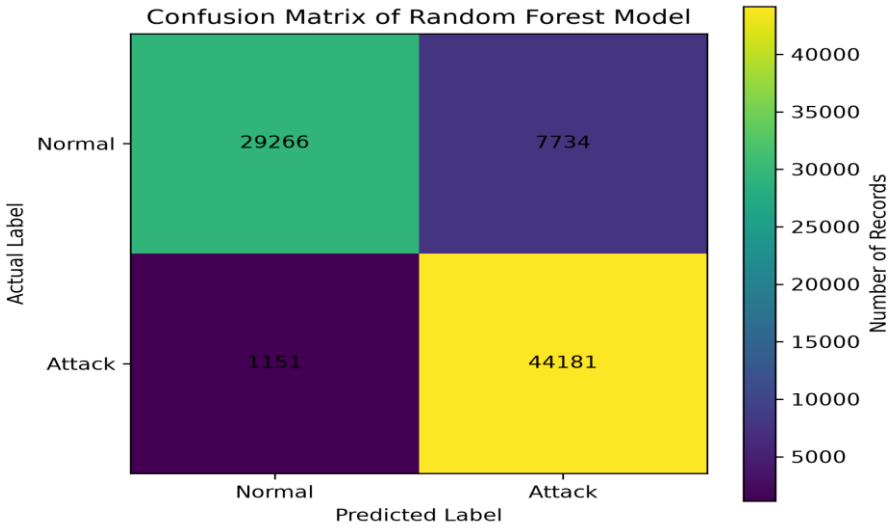

Figure 3. Confusion matrix of the Random Forest model.

4.3 Feature Importance Analysis

Feature importance analysis was performed using the Random Forest model to identify which network traffic characteristics contributed most strongly to classification. The most important feature was ct_state_ttl, which combines connection state and time-to-live-related information. The features sttl and dttl

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

also appeared among the most important predictors, showing that time-to-live values contributed significantly to classification.

Table 5. Top 15 important features identified by Random Forest

Rank	Feature	Importance
1	ct_state_ttl	0.082465
2	sttl	0.072172
3	sload	0.052479
4	rate	0.046801
5	dload	0.046420
6	dur	0.038257
7	sbytes	0.036320
8	ct_srv_dst	0.035120
9	dpkts	0.034324
10	smean	0.034143
11	dttl	0.033635
12	state_INT	0.031932
13	sinpkt	0.031203
14	synack	0.030795
15	tcprtt	0.029784

Traffic load and rate features, including sload, dload, and rate, were also influential. These features represent communication intensity and may help identify abnormal traffic bursts, scanning behavior, or attack flows. Flow duration, packet counts, byte counts, and TCP timing features also contributed to classification. The importance of these features confirms that the model relied on network flow behavior rather than application-layer web request characteristics.

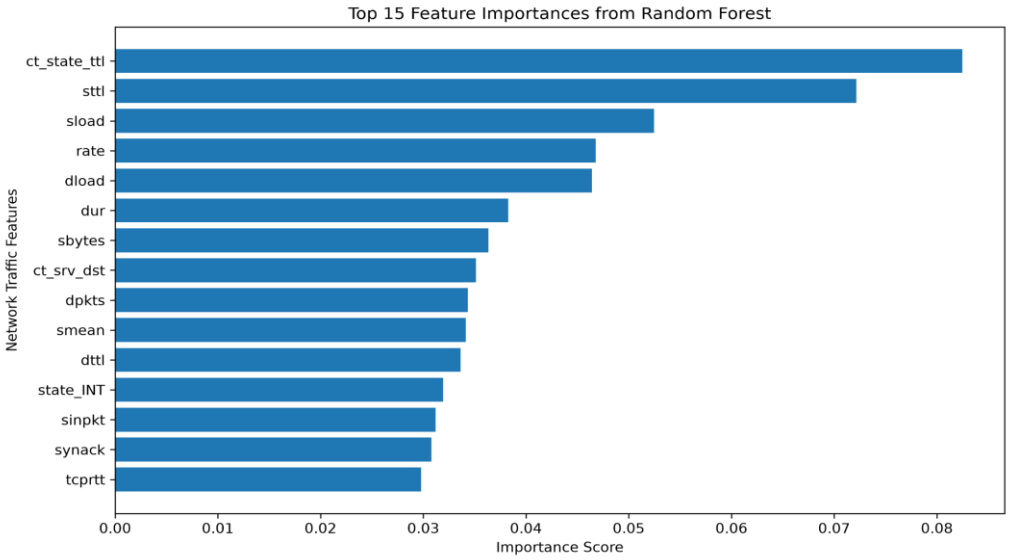

Figure 4. Top 15 feature importances from the Random Forest model.

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

5. Discussion

The experimental results demonstrate that machine learning models can support binary network flow classification for telecommunication cybersecurity. Random Forest achieved the strongest overall performance because network traffic classification depends on complex interactions among multiple flow-level features. Attack traffic may not be identified by a single variable alone, but by combinations of connection state, TTL values, packet counts, byte counts, flow duration, traffic rate, and load-related indicators.

The comparison between linear and tree-based models is important. Logistic Regression and Linear SVM produced acceptable results, but they were weaker than Decision Tree, Random Forest, and Gradient Boosting. Linear models are useful when classes can be separated by a relatively simple decision boundary. Network flow behavior is often nonlinear: for example, short duration alone may not indicate an attack, and high packet rate may also appear in legitimate traffic. Tree-based models are better suited for capturing such conditional interactions.

Gradient Boosting achieved the highest recall, meaning it detected the largest proportion of attack traffic. This may be useful in high-risk environments where missing attacks is more dangerous than investigating additional alerts. However, its precision was lower than that of Random Forest, suggesting more false positives. For balanced telecommunication cybersecurity monitoring, Random Forest is more suitable because it achieved the highest F1-score and better balance between precision and recall.

The findings are relevant to telecommunication cybersecurity because the study focuses on flow-level traffic rather than application-layer request content. Telecommunication networks process heterogeneous traffic from mobile users, enterprise systems, broadband customers, cloud services, and connected devices. Flow-level classification can support intrusion detection systems by identifying abnormal patterns across different protocols and services. It can also help security operations centers prioritize suspicious flows for further investigation.

However, machine learning-based classification should be used as a decision-support layer rather than a fully autonomous security decision-maker. Even the best model produced false positives and false negatives. In real environments, model outputs should be combined with rule-based intrusion detection, threat

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

intelligence, anomaly monitoring, and expert review. Continuous monitoring and periodic retraining are also necessary because network behavior changes over time due to new applications, devices, services, and attack techniques.

6. Limitations

This study has several limitations. First, the experiment used a public benchmark dataset rather than live telecommunication network traffic. Although UNSW-NB15 is suitable for controlled intrusion detection experiments, real telecom networks may contain different traffic patterns, routing conditions, service architectures, and attack strategies. Second, the task was formulated as binary classification. This is useful for initial detection, but multiclass classification would provide more detailed information for incident response and threat prioritization.

Third, the experiment used the available train-test partitions of the dataset. Additional validation strategies such as cross-validation, temporal validation, or testing on independent datasets could provide stronger evidence of model generalization. Fourth, the study evaluated classical machine learning models and did not include deep learning architectures. Fifth, feature importance was based on Random Forest impurity-based importance, which identifies influential variables but does not explain each individual prediction. Future work may use SHAP or LIME for deeper interpretability and may also evaluate real-time deployment constraints.

7. Conclusion

This study investigated machine learning-based binary network flow classification for telecommunication cybersecurity using the UNSW-NB15 dataset. Five supervised models were compared: Logistic Regression, Decision Tree, Random Forest, Linear Support Vector Machine, and Gradient Boosting. The study focused on flow-level network traffic features rather than web request-level indicators.

Random Forest achieved the best overall performance, with accuracy of 0.8921, precision of 0.8510, recall of 0.9746, and F1-score of 0.9086. Gradient Boosting achieved the highest recall of 0.9853, but its lower precision indicated a higher

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

false alarm tendency. The confusion matrix showed that Random Forest correctly detected 44,181 attack records and correctly classified 29,266 normal records, while producing 7,734 false positives and 1,151 false negatives.

Feature importance analysis showed that ct_state_ttl, sttl, sload, rate, dload, dur, sbytes, ct_srv_dst, dpkts, and smean were among the most influential predictors. These features relate to connection state, time-to-live values, traffic load, flow rate, duration, packet count, and byte volume. The findings suggest that tree-based models, especially Random Forest, can provide a useful balance between attack detection and false alarm reduction in benchmark-based network intrusion detection. Future research may extend this work through multiclass attack classification, testing on CICIDS2017 or real telecommunication traffic, explainable AI methods, and real-time deployment evaluation.

References

1. N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in Military Communications and Information Systems Conference, 2015.
2. UNSW Canberra, "The UNSW-NB15 Dataset," Australian Centre for Cyber Security, University of New South Wales. [Online]. Available: https://research.unsw.edu.au/projects/unsw-nb15-dataset
3. K. Scarfone and P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," NIST Special Publication 800-94, 2007. [Online]. Available: https://csrc.nist.gov/pubs/sp/800/94/final
4. A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1153-1176, 2016.
5. R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in Proceedings of the IEEE Symposium on Security and Privacy, pp. 305-316, 2010.
6. L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.
7. C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.

Modern American Journal of Engineering, Technology, and Innovation
ISSN(E): 3067-7939
Volume 2, Issue 5, May, 2026
Website: usajournals.org

8. J. H. Friedman, "Greedy function approximation: A gradient boosting machine," The Annals of Statistics, vol. 29, no. 5, pp. 1189-1232, 2001.
9. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.
10. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proceedings of ICISSP, pp. 108-116, 2018.
11. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in Proceedings of the IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009.
12. A. Thakkar and R. Lohiya, "A review of the advancement in intrusion detection datasets," Procedia Computer Science, vol. 167, pp. 636-645, 2020.
13. ENISA, "ENISA Threat Landscape 2024," European Union Agency for Cybersecurity, 2024. [Online]. Available: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2024
14. IBM Security, "Cost of a Data Breach Report 2024," IBM, 2024. [Online]. Available: https://www.ibm.com/reports/data-breach
15. International Telecommunication Union, "Facts and Figures 2024: Internet use," ITU, 2024. [Online]. Available: https://www.itu.int/itu-d/reports/statistics/2024/11/10/ff24-internet-use/