International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 6, Nov-Dec-2026, ISSN (Online): 2395-566X

Cyber Attack Prediction Using Machine Learning

Sonawane Manjusha Anil[1]
Department Of Artificial Intelligence and Data Science Sahyadri Valley College of Engineering and Technology, Rajuri, India[1]

Professor Divya Ashok Rahane[2]
Department Of Artificial Intelligence and Data Science Sahyadri Valley College of Engineering and Technology, Rajuri, India[2]

Abstract—The increasing frequency and complexity of cyber threats create a need for proactive security mechanisms that can identify suspicious network behaviour before it causes significant damage. This paper presents a machine-learning-based framework for cyber attack prediction using network traffic characteristics. The proposed approach collects network-flow records, performs data cleaning and feature preparation, and applies supervised learning algorithms to classify traffic as normal or potentially malicious. The framework considers benchmark datasets such as CICIDS2017 and UNSW-NB15, which contain normal traffic and multiple attack categories. Candidate models include Decision Tree, Random Forest, Support Vector Machine and gradient-boosting classifiers. Model performance is to be assessed using precision, recall, F1-score, confusion matrix and false-positive rate rather than accuracy alone. The architecture also includes an alert and reporting layer to support security analysts. The paper describes a proposed design and evaluation methodology; it does not claim experimental accuracy results because a completed, reproducible model evaluation is not supplied.

Keywords—Cyber Attack Prediction, Machine Learning, Intrusion Detection, Network Traffic Analysis, CICIDS2017, UNSW-NB15, Random Forest.

1. INTRODUCTION

Cyber attacks target the confidentiality, integrity and availability of digital systems. Common threats include denial-of-service (DoS), distributed denial-of-service (DDoS), port scanning, brute-force attempts, botnet activity, web attacks and exploitation of vulnerable services.

Traditional signature-based security tools are effective for known patterns, but they may fail to recognize previously unseen or slightly modified behaviours. Network intrusion detection systems (NIDS) therefore increasingly use statistical analysis and machine learning (ML) to identify deviations in traffic patterns [1], [2].

Cyber attack prediction in this paper refers to the early classification of observed network-flow behaviour as normal or suspicious, with the longer-term goal of supporting timely defensive action.

It should not be interpreted as certainty that a future attack will occur. A model estimates risk from available features and historical examples; the output must be interpreted alongside security context and analyst review.

The objectives are: (i) to design a practical ML pipeline for network-flow analysis; (ii) to compare suitable classification algorithms; (iii) to identify useful evaluation measures for imbalanced security data; and (iv) to present an architecture that can generate understandable alerts.

II. RELATED WORK

Research on intrusion detection has explored statistical methods, data mining, classical ML and deep learning. Khraisat et al. [1] discuss the major IDS approaches, datasets and challenges, including the difficulty of detecting new attacks while controlling false alarms. Buczak and Guven [2] review data-mining and ML techniques used in cyber intrusion analytics. These works emphasize that dataset quality, feature selection and evaluation design strongly influence reported performance.

Sharafaldin et al. introduced CICIDS2017 as a contemporary intrusion-detection benchmark with benign traffic and several attack scenarios [3]. Moustafa and Slay developed UNSW-NB15 to represent modern normal and malicious network activity [4]. Both datasets are widely used for comparative IDS research, but differences in traffic generation, labels and class distributions mean that results should not be compared without considering the experimental setup.

More recent studies investigate ML and deep-learning approaches across benchmark datasets. A survey published in 2024 summarizes the use of datasets including NSL-KDD, UNSW-NB15 and CICIDS2017, and discusses methodological considerations in ML-based IDS [5]. Studies of cyberattack prediction and IoT security further show the value of combining traffic features with careful validation [6], [7].

© 2026 IJSRET

III. PROPOSED SYSTEM

1. **System Overview**

The proposed system is a supervised learning pipeline that receives network-flow records and produces a predicted class and risk indication. The input may be exported flow records from a network monitor or a labelled benchmark dataset. Raw packet payload inspection is outside the scope of this proposed flow-based design. The system is intended as a decision-support layer and does not automatically block traffic.

2. **System Architecture**

The architecture is organized into six stages: traffic/data input, preprocessing, feature engineering, model training or inference, risk classification, and alert/report generation. During training, labelled records are split into training, validation and test partitions. During inference, the same fitted preprocessing operations are applied to new records before classification.

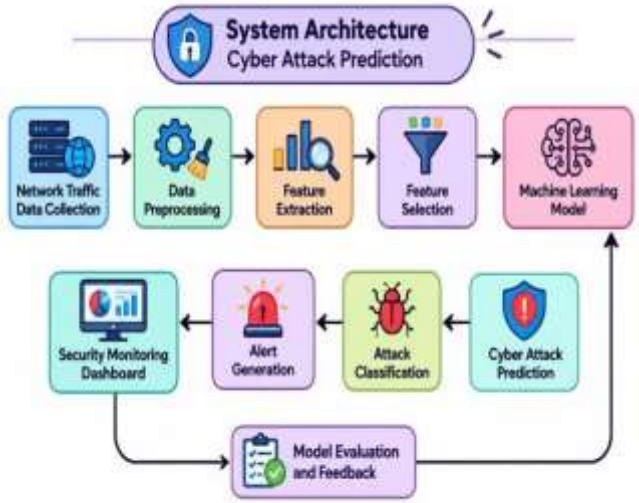

Fig. 1. Proposed system architecture for machine-learning-based cyber attack prediction.

3. Dataset Selection

CICIDS2017 and UNSW-NB15 are proposed as benchmark sources. CICIDS2017 contains benign traffic and multiple attack scenarios such as DoS, DDoS, port scan, brute force, bot and web attacks [3]. UNSW-NB15 contains normal traffic and attack families such as fuzzers, analysis, backdoors, DoS, exploits, generic activity, reconnaissance, shellcode and worms [4]. Dataset documentation should be checked before downloading because versions and processed copies may differ.

Table I. Candidate Benchmark Datasets

Dataset	Typical coverage	Use in this study
CICIDS2017	Benign traffic; DoS/DDoS, port scan, brute force, bot, web and infiltration attacks	Primary candidate for flow-based model development
UNSW-NB15	Normal traffic; nine broad attack families	Independent benchmark / cross-dataset robustness check

4. Data Preprocessing

The data preparation stage checks column names and data types, removes duplicate rows, converts infinite values to missing values, and handles missing entries using a documented strategy. Categorical fields such as protocol or service are encoded. Numerical features may be standardized for algorithms sensitive to feature scale, such as Support Vector Machines. Identifiers that could leak labels or cause memorization should be reviewed and excluded when inappropriate. The target label is mapped to binary classes for normal-versus-attack prediction; a second multi-class experiment can be used to identify attack families.

Table 2. Candidate Machine Learning Algorithms

Algorithm	Strength	Consideration
Decision Tree	Easy to interpret and explain	May overfit without pruning or depth limits
Random Forest	Ensemble robustness; handles non-linear patterns	Larger model and less transparent globally
SVM	Useful for high-dimensional feature spaces	Scaling and parameter tuning are important
Gradient Boosting	Captures complex feature interactions	Needs careful tuning and leakage-safe validation

5. Machine Learning Algorithms

Decision Tree provides an interpretable baseline by learning feature-based decision rules. Random Forest combines multiple decision trees and can model non-linear relationships while reducing the variance of a single tree [8]. Support Vector Machine can separate classes using a decision boundary and may be effective after appropriate scaling [9]. Gradient-boosting methods build an ensemble sequentially and can provide strong tabular-data

© 2026 IJSRET

performance [10]. A simple baseline should be retained so that improvements can be judged fairly.

IV. METHODOLOGY AND EVALUATION

1. Experimental Workflow

First, obtain the official dataset and record its version, source and licence. Second, inspect label distribution and remove malformed records. Third, separate features from labels and create stratified train, validation and test partitions. Where the dataset provides a recommended split, preserve it for a separate reproducibility experiment. All preprocessing steps must be fitted only on the training partition and then applied to validation and test data to prevent data leakage.

Fourth, train the candidate algorithms using consistent partitions. Hyperparameters are selected using validation data or cross-validation confined to the training portion. Fifth, evaluate the final selected model once on the held-out test set. For cross-dataset testing, train on one dataset and evaluate on the other only after aligning compatible feature definitions; incompatible features must not be silently mapped.

2. Performance Metrics

Accuracy alone can be misleading when normal and attack records are imbalanced. Precision measures how many predicted attacks are truly attacks; recall measures how many actual attacks are detected. The F1-score is the harmonic mean of precision and recall. The false-positive rate (FPR) is important because excessive false alarms can overwhelm security teams. Confusion matrices and per-class metrics should be reported for multi-class classification.

Table III. Performance Evaluation Metrics

Metric	Formula / meaning	Why it matters
Accuracy	(TP + TN) / (TP + TN + FP + FN)	Overall proportion classified correctly
Precision	TP / (TP + FP)	Controls false attack alerts
Recall	TP / (TP + FN)	Measures missed attacks
F1-score	2 × (Precision × Recall) / (Precision + Recall)	Balances precision and recall
FPR	FP / (FP + TN)	Measures normal traffic incorrectly flagged

3. Expected Output

For each input flow, the system is expected to display the predicted label, a model confidence or risk indicator where calibrated, and selected feature information that helps an analyst review the event. The dashboard may summarize total records, normal records, suspicious records and predicted attack categories. These are proposed interface outputs, not measured results.

4. Validation and Reliability

The evaluation should consider class imbalance, duplicate flows, temporal correlation and possible train-test contamination. Random splitting may place highly similar flows in both training and testing sets and overstate generalization. A time-aware split is preferable when timestamps are available. Threshold selection should be documented, and performance should be checked at more than one operating threshold. Model drift should be monitored because legitimate traffic and attack techniques change over time.

5 Applications and Benefits

The proposed framework can support enterprise network monitoring, campus network security, data-centre traffic analysis, cloud environments, Internet of Things (IoT) deployments and security-operation-centre triage. It can help prioritize suspicious events, identify recurring attack patterns and reduce dependence on manually written signatures. Its modular design allows a dataset, classifier or reporting interface to be replaced without redesigning the complete pipeline.

The main benefits are automated analysis of large flow records, repeatable model comparison, support for multi-class attack categorization, and metric-based assessment. However, predictions are not proof of malicious intent. An analyst should verify high-risk events before taking disruptive action.

6 Limitations and Future Scope

Public benchmark datasets are collected in controlled environments and may not represent every organization's live traffic. Label errors, class imbalance, outdated attack patterns and feature differences can affect performance. Encrypted traffic limits visibility into payload content, while an ML model may produce false positives or fail on zero-day behaviours not represented in training data. Model outputs can also be affected by adversarial manipulation.

Future work may include real-time streaming ingestion, online or incremental learning, explainable AI methods, calibration of risk scores, federated learning across organizations, and evaluation on newer IoT and enterprise datasets. A production deployment would additionally

© 2026 IJSRET

require access control, secure logging, privacy safeguards and human approval for response actions.

V. CONCLUSION

This paper presented a proposed machine-learning framework for cyber attack prediction based on network-flow characteristics. The design combines data preprocessing, feature engineering, supervised classification, performance evaluation and analyst-oriented alerting. CICIDS2017 and UNSW-NB15 provide suitable benchmark starting points, while Decision Tree, Random Forest, SVM and gradient boosting offer a practical set of candidate models. Reliable assessment requires leakage-aware data splitting and metrics such as precision, recall, F1-score and false-positive rate. The framework provides a foundation for implementation and experimental validation; actual performance must be established through reproducible testing on the selected datasets.

Acknowledgment
The authors express their sincere gratitude to the Department of Artificial Intelligence and Data Science, Sahyadri Valley College of Engineering and Technology, Rajuri, for academic guidance and support.

REFERENCES

1. A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 2, article 20, 2019.
2. 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.
3. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. 4th International Conference on Information Systems Security and Privacy (ICISSP), 2018, pp. 108–116.
4. N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems," in Proc. Military Communications and Information Systems Conference (MilCIS), 2015, pp. 1–6.
5. "Unveiling machine learning strategies and considerations in intrusion detection systems: a comprehensive survey," Frontiers in Computer Science, 2024, doi: 10.3389/fcomp.2024.1387354.
6. "Conceptualisation of cyberattack prediction with deep learning," Cybersecurity, 2020, doi: 10.1186/s42400-020-00053-7.
7. "Cyberattacks detection in IoT-based smart city applications using machine learning techniques," Sensors, 2020.
8. L. Breiman, "Random forests," Machine Learning, vol. 45, pp. 5–32, 2001.
9. C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, pp. 273–297, 1995.
10. J. H. Friedman, "Greedy function approximation: a gradient boosting machine," The Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001.
11. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in Proc. IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009, pp. 1–6.
12. R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Applying convolutional neural network for network intrusion detection," in Proc. International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 1222–1228.

AUTHOR PROFILE



SONAWANE MANJUSHA ANIL
Student, Department of Artificial Intelligence and Data Science, Sahyadri Valley College of Engineering and Technology, Rajuri, India. Research interests include machine learning, network security and cyber attack prediction.



PROF. DIVYA ASHOK RAHANE
Faculty, Department of Artificial Intelligence and Data Science, Sahyadri Valley College of Engineering and Technology, Rajuri, India. Email: divyarahane26@gmail.com.

© 2026 IJSRET