



DOI: <https://doi.org/10.38035/gcir.v1i2>
<https://creativecommons.org/licenses/by/4.0/>

Received: 01 September 2025 | Revised: 07 September 2025 | Accepted: 20 September 2025

A Systematic Review of Machine Learning Methods for Network Intrusion Detection: Performance, Datasets, and Cybersecurity Implementation Challenges

Ridwan¹, Fivi Aprilianingsih²

¹Sekolah Tinggi Teknologi Pekerjaan Umum, Jakarta, Indonesia, ridwansamm@sttpu.ac.id

²Universitas Bhayangkara Jakarta Raya, Jakarta, Indonesia, aprilianingsihfivi@gmail.com

Corresponding Author: ridwansamm@sttpu.ac.id¹

Abstract: The increasing frequency and complexity of cyberattacks have highlighted the need for effective Network Intrusion Detection Systems (NIDS). Machine learning (ML) has emerged as a promising approach for identifying malicious network traffic and detecting anomalous activities. However, differences in algorithms, datasets, evaluation metrics, and experimental settings make it difficult to compare existing approaches and determine their suitability for real-world deployment. This study aims to systematically review machine learning methods for network intrusion detection, with particular emphasis on model performance, dataset characteristics, evaluation practices, and implementation challenges. The review adopts a Systematic Literature Review (SLR) approach guided by the PRISMA 2020 reporting framework. The analysis focuses on machine learning and deep learning techniques, commonly used benchmark datasets, and evaluation metrics such as accuracy, precision, recall, F1-score, and false-positive rate. The reviewed literature indicates that ensemble learning and hybrid models demonstrate promising detection performance on several benchmark datasets, while deep learning approaches provide opportunities to identify complex traffic patterns and network anomalies. Nevertheless, high performance on benchmark datasets does not necessarily guarantee effective deployment in operational environments. Key challenges include class imbalance, false-positive alarms, limited cross-dataset generalization, computational overhead, and the detection of rare or previously unseen attacks. This review emphasizes the need for transparent experimental reporting, cross-dataset evaluation, multiple performance metrics, and consideration of computational efficiency when assessing NIDS models. The findings provide a structured perspective for researchers and cybersecurity practitioners in selecting, evaluating, and developing more reliable machine learning-based intrusion detection systems.

Keywords: Machine Learning, Network Intrusion Detection, Cybersecurity, Systematic Literature Review, Deep Learning, Intrusion Detection System.

INTRODUCTION

The rapid development of information and communication technologies has increased the reliance of organizations on computer networks to support data exchange, digital services, and business operations. However, the growing connectivity of network infrastructures has also expanded the potential attack surface for cyber threats. Network-based attacks, including denial-of-service attacks, port scanning, botnet activities, and unauthorized access, pose significant risks to the confidentiality, integrity, and availability of information systems. Consequently, effective network security mechanisms are essential for identifying malicious activities and mitigating potential threats. Intrusion detection systems (IDSs) have become an important component of cybersecurity infrastructure because they monitor network traffic or system activities to identify suspicious behavior and potential security violations (Mishra et al., 2019; Shaukat et al., 2020).

Traditional intrusion detection approaches generally rely on predefined rules, signatures, or manually developed detection patterns. Although these approaches can effectively identify known attack signatures, their ability to detect previously unseen attacks and evolving threat patterns may be limited. Machine learning (ML) has therefore attracted considerable attention as an alternative approach to intrusion detection. By learning patterns from network traffic and historical security data, ML-based intrusion detection systems can support the identification of malicious activities, anomaly detection, and the classification of network traffic into normal and attack categories (Saranya et al., 2020; Kilincer et al., 2021). The application of artificial intelligence to cybersecurity has also been investigated in broader contexts, including Internet of Things (IoT) environments and critical infrastructure, where increasing device connectivity introduces additional security challenges (Abdullahi et al., 2022; Pinto et al., 2023).

A wide range of machine learning algorithms has been investigated for network intrusion detection. These algorithms include decision trees, random forests, support vector machines (SVMs), k-nearest neighbors (k-NN), naïve Bayes, artificial neural networks, and more advanced deep learning approaches. Each method offers different advantages in terms of classification performance, computational requirements, interpretability, and suitability for particular data characteristics. Ensemble methods, for example, combine predictions from multiple models, whereas deep learning approaches can learn complex representations from large datasets. Nevertheless, no single algorithm can be assumed to perform optimally across all network environments and attack scenarios. Model effectiveness depends on several factors, including dataset characteristics, feature selection, preprocessing procedures, class distributions, and experimental design (Mishra et al., 2019; Shaukat et al., 2020; Stewart et al., 2024).

The reliability of performance comparisons remains a major methodological concern in intrusion detection research. Studies may employ different datasets, feature representations, preprocessing strategies, training–testing partitions, and evaluation protocols. Consequently, reported performance values may not be directly comparable across studies. A model that achieves high accuracy on one benchmark dataset may perform differently when applied to another dataset or to network traffic collected under different operational conditions. Research on anomaly-based intrusion detection has highlighted the importance of factors such as class imbalance and potential discrepancies between training and testing data when evaluating machine learning models (Abdelmoumin et al., 2021). These issues indicate that performance should be interpreted in relation to the experimental conditions under which a model was evaluated.

Dataset selection is particularly important because the characteristics of the data influence the patterns that a model can learn. Publicly available datasets, including CICIDS2017, CSE-CIC-IDS2018, and UNSW-NB15, are frequently considered in intrusion detection research. However, differences in attack categories, traffic characteristics, data collection procedures, feature availability, and class distributions can affect experimental

outcomes. A dataset may provide useful evidence for comparing algorithms under controlled conditions, but it does not necessarily represent the diversity and complexity of traffic in operational networks. Therefore, the use of benchmark datasets should be examined alongside the limitations of the data and the extent to which experimental findings can be generalized to different network environments (Kilincer et al., 2021; Stewart et al., 2024).

Another important issue concerns the selection and interpretation of evaluation metrics. Accuracy is widely used to report classification performance, but it may provide an incomplete picture when the dataset contains imbalanced classes. In such circumstances, a model may achieve high overall accuracy while failing to detect a substantial proportion of attacks belonging to minority classes. Precision, recall, and the F1-score can provide additional information about detection effectiveness, while the false-positive rate is relevant to understanding the burden of incorrectly identified malicious activities. For operational applications, computational cost, inference latency, and resource requirements may also be important when these measures are reported by the original studies. A comprehensive assessment of ML-based intrusion detection should therefore consider multiple evaluation metrics rather than relying exclusively on accuracy (Abdelmoumin et al., 2021; Saranya et al., 2020).

Beyond predictive performance, the practical implementation of machine learning-based intrusion detection systems presents additional challenges. Network environments change over time, and attack techniques continue to evolve. Models trained on historical traffic may consequently require monitoring, updating, or retraining to maintain their effectiveness. The computational demands of some methods may also affect their suitability for environments requiring low-latency detection or limited processing resources. Furthermore, the availability and quality of training data, the selection of relevant features, and the possibility of adversarial manipulation can influence the reliability of the resulting detection system. Previous research has discussed these issues in the broader context of machine learning for cybersecurity, indicating that successful intrusion detection requires attention to both algorithmic performance and implementation conditions (Shaukat et al., 2020; Harahsheh & Chen, 2023; Stewart et al., 2024).

Several previous studies have reviewed machine learning applications in intrusion detection and cybersecurity. Shaukat et al. (2020) surveyed machine learning techniques for cybersecurity, while Saranya et al. (2020) reviewed the performance of machine learning algorithms in intrusion detection systems. Kilincer et al. (2021) examined machine learning methods, datasets, and comparative performance in cybersecurity intrusion detection. Other reviews have addressed artificial intelligence-based cybersecurity in IoT environments (Abdullahi et al., 2022), intrusion detection for critical infrastructure (Pinto et al., 2023), and machine learning-based intrusion detection systems more broadly (Stewart et al., 2024). These contributions provide important foundations for understanding the development of the field. However, the continuing diversity of datasets, evaluation metrics, experimental procedures, and implementation requirements makes it necessary to critically examine how reported results can be interpreted across studies.

Accordingly, the contribution of the present review is not based on the assumption that no previous systematic review exists. Instead, it focuses on an integrated synthesis of machine learning methods, dataset characteristics, evaluation metrics, and cybersecurity implementation challenges in network intrusion detection. By examining these dimensions together, the review aims to clarify the relationship between reported model performance and the experimental or operational conditions under which that performance is obtained. Particular attention is given to class imbalance, false-positive rates, generalization across datasets, and computational considerations where the primary studies provide relevant evidence.

This study addresses the following research questions:

- 1) RQ1: What machine learning methods have been investigated for network intrusion detection, and how are these methods applied?
- 2) RQ2: Which datasets and evaluation metrics are used to assess the performance of machine learning-based network intrusion detection systems?
- 3) RQ3: What methodological and implementation challenges are reported, particularly regarding class imbalance, false-positive rates, generalization, and computational requirements?
- 4) RQ4: To what extent do results obtained from public benchmark datasets provide evidence of effectiveness in real-world network environments?

To address these questions, this study adopts a systematic literature review (SLR) approach. The review process will establish explicit search strategies, eligibility criteria, study-selection procedures, and data-extraction fields. The reporting of the review will be guided by the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA 2020) statement to improve transparency and reproducibility (Page et al., 2021). The literature search will focus on studies published between 2020 and 2026, with the final search date specified in the methodology. Earlier publications may be considered where necessary to explain foundational methods or provide relevant context. The synthesis will combine descriptive analysis of research characteristics with a thematic comparison of algorithms, datasets, metrics, and implementation challenges. A quantitative meta-analysis will only be considered if the included studies are sufficiently comparable in their methods and reported outcomes.

The expected contribution of this review is a structured account of the current evidence on machine learning-based network intrusion detection, emphasizing not only reported predictive performance but also the conditions and limitations associated with that performance. The findings are expected to help researchers identify methodological gaps, select appropriate evaluation practices, and formulate future research directions. For practitioners, the review may provide a basis for interpreting benchmark results more cautiously and assessing the considerations that influence the implementation of intrusion detection systems in operational cybersecurity environments.

METHODS

Research Design

This study adopts a Systematic Literature Review (SLR) to examine machine learning methods for network intrusion detection, focusing on model performance, dataset characteristics, evaluation metrics, and cybersecurity implementation challenges. An SLR provides a structured approach to identifying, selecting, evaluating, and synthesizing research evidence. The review is guided by the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA 2020) statement (Page et al., 2021). PRISMA will be used to report study selection transparently; it does not replace the review protocol or methodological appraisal.

The review combines descriptive analysis of study characteristics with thematic synthesis. The primary unit of analysis is a research publication reporting empirical results, methodological findings, or substantive evaluations of machine learning-based network intrusion detection systems (NIDS). IoT-related studies will be considered when their findings directly concern network traffic analysis or intrusion detection and meet the eligibility criteria.

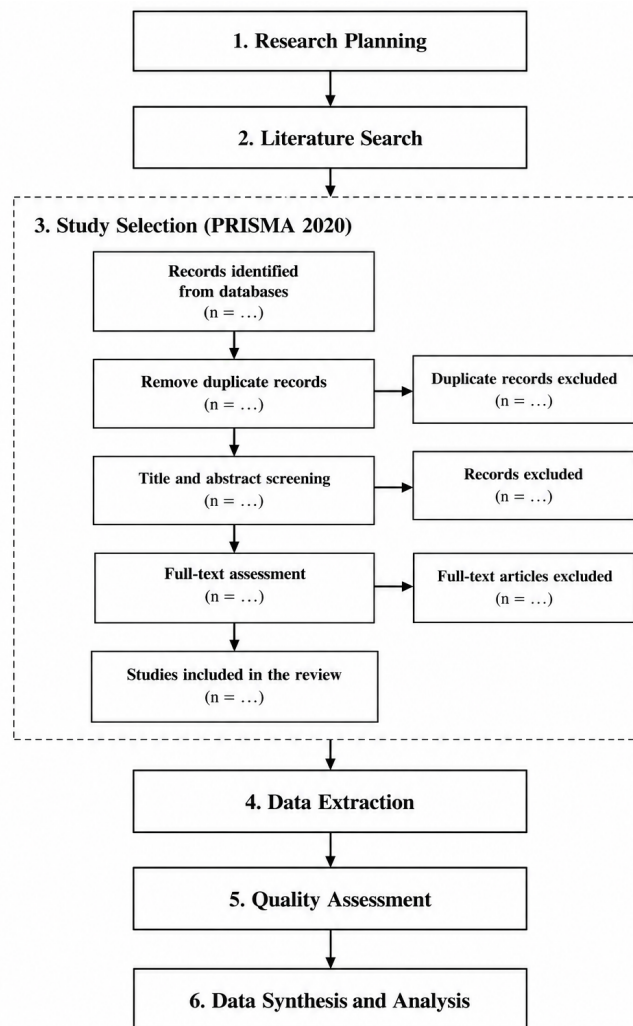


Figure 1. Research design of the systematic literature review

Research Questions

The review addresses the following research questions:

- 1) RQ1: What machine learning methods have been investigated for network intrusion detection, and how are these methods applied?
- 2) RQ2: Which datasets and evaluation metrics are used to assess the performance of machine learning-based network intrusion detection systems?
- 3) RQ3: What methodological and implementation challenges are reported, particularly regarding class imbalance, false-positive rates, generalization, and computational requirements?
- 4) RQ4: To what extent do results obtained from public benchmark datasets provide evidence of effectiveness in real-world network environments?

These questions guide the search strategy, eligibility criteria, data extraction, and synthesis, linking reported model performance to the experimental and operational conditions under which it was obtained.

Literature Search Strategy

A systematic search will be conducted in Scopus, Web of Science, IEEE Xplore, the ACM Digital Library, and ScienceDirect, subject to access availability. These databases are intended to cover peer-reviewed research in computer science, machine learning, network security, and intrusion detection.

The primary publication period will cover January 2020 through the final search date in 2026. The final search date will be reported because 2026 is still ongoing. Earlier publications may be cited selectively for foundational context but will not automatically be included in the primary synthesis.

The preliminary search string is: ("intrusion detection" OR "network intrusion detection" OR NIDS OR "intrusion detection system") AND ("machine learning" OR "deep learning" OR "supervised learning" OR "unsupervised learning" OR "ensemble learning") AND (network OR cybersecurity OR "network traffic"). Search syntax will be adapted to each database. Supplementary searches may add terms related to datasets, evaluation metrics, class imbalance, generalization, and real-time deployment.

For reproducibility, the database name, exact search string, search date, filters, and number of retrieved records will be recorded. Records will be consolidated and duplicates identified before screening. Actual search results and counts will be reported only after the searches have been performed.

Eligibility Criteria

Predefined inclusion and exclusion criteria will be applied during title-and-abstract screening and full-text assessment. Publications whose relevance is uncertain during initial screening will be retained for full-text assessment.

Table 1. Eligible Criteria

Dimension	Inclusion criteria	Exclusion criteria
Research focus	ML-based network intrusion detection or relevant security traffic classification.	Unrelated to intrusion detection, network security, or the research questions.
Methodological relevance	Evaluates relevant ML methods, datasets, metrics, or implementation challenges.	Mentions ML without relevant methodological or analytical information.
Publication period	2020 through the final search date in 2026.	Outside the period, except selected background references.
Publication type	Peer-reviewed journal articles and full conference papers with sufficient detail.	Editorials, abstracts without full papers, duplicates, or insufficiently documented publications.
Language	English-language publications.	Publications in other languages under this review's scope.
Availability	Full text available and sufficient information to assess and extract data.	Full text unavailable after reasonable retrieval attempts or insufficient information.

Source: Research Studies

Study Selection Process

Study selection will follow the identification, screening, eligibility, and inclusion stages described in PRISMA 2020 (Page et al., 2021). Records from all databases will be consolidated and duplicates removed. Titles and abstracts will then be screened against the eligibility criteria, followed by full-text assessment of potentially relevant publications.

The PRISMA flow diagram will report the actual number of records identified, duplicates removed, records screened, full texts assessed, and studies included. Reasons for full-text exclusion will be documented. No counts will be specified before searching and screening are completed. If multiple reviewers participate, disagreements will be resolved through discussion and, where needed, a third reviewer. If one researcher conducts the review, this limitation will be acknowledged and decisions documented systematically.

Data Extraction

A structured data-extraction form will be developed and piloted on a small subset of eligible publications. The following information will be extracted where reported:

- 1) Bibliographic information: authors, year, title, publication venue, and DOI or persistent identifier.
- 2) Research characteristics: objectives, application context, network environment, and detection task.
- 3) Machine learning methods: algorithms, model configurations where available, and supervised, unsupervised, semi-supervised, ensemble, or deep learning approaches.
- 4) Dataset characteristics: dataset name and source, attack categories, sample characteristics where reported, and public or private traffic source.
- 5) Experimental procedures: preprocessing, feature selection, class balancing, data partitioning, validation strategy, and baseline comparisons.
- 6) Performance metrics: accuracy, precision, recall, F1-score, false-positive rate, and latency or computational cost where reported.
- 7) Implementation considerations: data quality, class imbalance, generalization, resource requirements, real-time detection, model maintenance, and deployment conditions.
- 8) Principal findings and limitations reported by the study.

Only information supported by the original publication will be recorded. Missing information will be marked as “not reported” rather than inferred. Differences in metric definitions and experimental settings will be preserved and considered during synthesis.

Methodological Quality and Risk-of-Bias Assessment

Included studies will be assessed using a structured appraisal checklist developed for this review. The checklist will examine dataset descriptions, preprocessing, experimental design, evaluation metrics, validation strategy, and limitations, as well as whether conclusions are supported by the reported results.

Reporting completeness will be distinguished from methodological robustness. Each applicable criterion will be recorded as satisfied, partially satisfied, not satisfied, or not reported. The assessment will contextualize findings and identify recurring weaknesses. A low appraisal result will not automatically exclude a study unless an exclusion rule has been predefined and justified.

Data Synthesis and Analysis

The evidence will be synthesized using descriptive and thematic analysis. A quantitative meta-analysis will only be considered if the studies are sufficiently comparable in datasets, attack categories, features, training procedures, validation strategies, and metric definitions. Otherwise, combining performance estimates could be misleading.

For RQ1, methods will be classified into conventional supervised algorithms, unsupervised methods, ensemble techniques, and deep learning. For RQ2, datasets and evaluation metrics will be mapped, interpreting performance in light of dataset characteristics and experimental procedures. Latency and computational-resource measurements will be summarized separately when available.

For RQ3, thematic synthesis will identify challenges such as class imbalance, false-positive rates, data quality, feature selection, overfitting, cross-dataset generalization, computational requirements, and real-time deployment. Additional themes may be added when supported by the extracted evidence.

For RQ4, the review will examine evidence beyond single-benchmark evaluation, including multi-dataset testing, cross-dataset evaluation, independently collected traffic, temporal validation, and deployment-oriented experiments. Findings will be presented in tables

and narrative form. Studies with substantially different experimental conditions will be described separately rather than ranked by performance values alone.

Transparency and Reproducibility

The final review will report the databases searched, search dates, complete search strings, eligibility criteria, study-selection counts, exclusion reasons, extraction fields, and synthesis procedures. The PRISMA flow diagram will document study selection, and summary tables will link included publications to synthesized findings. Protocol changes will be documented and justified. Findings directly reported by primary studies will be distinguished from interpretations developed through cross-study synthesis.

RESULTS AND DISCUSSION

Search and Study Selection

A focused web-based literature search was conducted on 24 October 2026 using publisher and scholarly repository pages, including SpringerLink, SpringerOpen, MDPI, Nature Portfolio, and PubMed Central. Search terms combined network intrusion detection (NIDS/IDS), machine learning or deep learning, benchmark datasets, class imbalance, feature selection, cross-dataset generalization, and deployment. Example queries included: ("network intrusion detection" OR NIDS) AND ("machine learning" OR "deep learning"); intrusion detection AND (CICIDS2017 OR UNSW-NB15 OR NSL-KDD OR CSE-CIC-IDS2018); and (class imbalance OR oversampling OR generalization) AND intrusion detection.

Candidate records were screened for relevance to network or network-related intrusion detection, use of a machine-learning method, reporting of quantitative evaluation or substantive methodological evidence, publication in 2020–2026, and availability of a publisher or repository record. Fifteen studies were retained for the evidence matrix below. The year 2026 is ongoing at the search date.

Important limitation: this was a focused search of accessible publisher pages, not a complete export-driven search through institutional Scopus, Web of Science, IEEE Xplore, and ACM Digital Library interfaces. Therefore, the 15 papers form a bounded evidence set rather than an exhaustive census. The final manuscript should rerun searches in the databases listed in Methodology, deduplicate exported records, document full-text exclusion reasons, and generate PRISMA counts from actual records. No comprehensive database-level counts are claimed here.

Characteristics of Selected Studies

Table 2. Characteristics of Studies

No.	Study	Method / emphasis	Dataset(s) / focus
1	Optimization of predictive performance of intrusion detection system using hybrid ensemble model for secure systems (2023)	Random Forest + recursive feature elimination	NSL-KDD, UNSW-NB15, CSE-CIC-IDS2018
2	Sajid et al. (2024), Enhancing intrusion detection: a hybrid machine	XGBoost/CNN feature selection + LSTM	CICIDS2017, UNSW-NB15, NSL-KDD, WSN-DS

	and deep learning approach			
3	Alars & Kurnaz (2024), Enhancing NIDS with combined network and host traffic features	CNN, selection, reduction	feature dimensionality	Military-network intrusion dataset
4	Ayantayo et al. (2023), Network intrusion detection using feature fusion with deep learning	Early fusion, fusion, late-ensemble	late	UNSW-NB15, NSL-KDD
5	Ren et al. (2023), A lightweight unsupervised intrusion detection model based on VAE	Lightweight spectral permutation entropy	VAE, residual,	SWaT industrial-control data
6	Sri Abhijit et al. (2025), Federated transfer learning for rare attack class detection	Federated learning, layers	transfer personalized client	CSE-CICIDS-2018, UNSW-NB15, Edge-IIoT
7	Al-Ghuwairi et al. (2023), Intrusion detection in cloud computing based on time series anomalies	Prophet and collaborative selection	forecasting feature	Cloud anomaly time series
8	Nazir et al. (2023), A novel feature-selection algorithm in IoT networks for intrusion detection	CAT-S selection + Random Forest	feature	TON_IoT
9	Talukder et al. (2024), ML-based NIDS for big and imbalanced data	Random oversampling, feature embedding, PCA	stacking	UNSW-NB15, CICIDS2017, CSE-CIC-IDS2018
10	Analysis of autoencoders for network intrusion detection (2021)	Autoencoder-based anomaly detection		Benchmark NIDS data; model sensitivity
11	Addressing class imbalance in NIDS using data resampling and deep learning (2023)	Resampling and learning	deep	NIDS benchmark datasets

12	Cantone et al. (2024), On cross-dataset generalization of ML for NIDS	Cross-dataset classifier evaluation	Multiple NIDS datasets; distribution shift
13	More et al. (2024), Enhanced intrusion detection systems performance with UNSW-NB15 data analysis	Comparative/ensemble ML analysis	UNSW-NB15
14	Pospíchal et al. (2026), Random Forest-based NIDS with feature selection and class balancing	Random Forest, mutual information, SMOTE	UNSW-NB15; attack-specific binary tasks
15	FedPrIDS (2026), privacy-preserving federated learning for collaborative NIDS in IoT	Federated DNN and differential privacy	CICIDS2017, UNSW-NB15, Bot-IoT

Source : Research Studies

Machine Learning Methods and Reported Performance (RQ1)

The selected studies cover tree-based ensembles, hybrid machine/deep-learning architectures, feature-fusion networks, unsupervised autoencoders, time-series anomaly detection, and federated learning. Tree-based ensembles remain attractive where feature selection and computational efficiency matter. The RF-RFE study reports overall accuracy of 99.0% on NSL-KDD, 98.53% on UNSW-NB15, and 99.9% on CSE-CIC-IDS2018. Talukder et al. (2024) combine oversampling, feature embedding, and dimensionality reduction, reporting accuracy up to 99.95% for Extra Trees on UNSW-NB15 and up to 99.99% for selected tree models on CICIDS2017.

Hybrid models combine components with complementary roles. Sajid et al. (2024) use XGBoost or CNN for feature selection and LSTM for classification across four datasets. Ayantayo et al. (2023) compare early-fusion, late-fusion, and late-ensemble deep networks and report that late-fusion and late-ensemble approaches show better generalization behaviour against overfitting. Such designs can improve detection but add model complexity and computational cost.

Unsupervised models address settings where attack labels are scarce. Ren et al. (2023) report an F1-score of 84.81% for a lightweight VAE-based model on the SWaT industrial-control dataset, with advantages in time and memory overhead. This score should not be ranked directly against supervised accuracy values because the dataset, task, metric, and operational setting differ. Overall, the evidence does not establish a universally superior algorithm: reported results depend on dataset, task framing, feature engineering, class distribution, and validation protocol.

Datasets and Evaluation Metrics (RQ2)

CICIDS2017, CSE-CIC-IDS2018, UNSW-NB15, and NSL-KDD recur across the selected literature. These datasets facilitate reproducible benchmarking, but studies differ in binary versus multiclass framing, attack subsets, preprocessing, selected features, class balancing, and train/test splits. Consequently, headline accuracy values from different studies should not be treated as directly comparable.

Accuracy should be interpreted alongside precision, recall, F1-score, false-positive rate, and class-specific confusion matrices. Talukder et al. (2024) report very high accuracy after oversampling and feature processing; the result therefore needs to be read in light of the entire pipeline. Pospíchal et al. (2026) found that SMOTE effects varied by attack category: some F1-scores improved while others declined, and some class-specific false-positive trade-offs emerged. This indicates that resampling should be evaluated separately for each attack class and against the original imbalanced distribution.

Domain-specific datasets also influence interpretation. SWaT represents an industrial-control environment, TON_IoT is IoT-oriented, and Edge-IIoT represents edge/IoT scenarios. Results from these environments provide useful evidence for those contexts but should not be generalized automatically to enterprise networks.

Class Imbalance

The reviewed approaches address class imbalance through random oversampling, SMOTE, feature selection, class-aware architectures, and deep learning. These techniques do not produce uniform benefits. Oversampling can increase the representation of rare classes but may also create synthetic patterns or alter decision boundaries. The attack-dependent SMOTE results reported on UNSW-NB15 support reporting per-class precision, recall, F1-score, confusion matrices, and pre-/post-resampling results.

False Positives

False positives are operationally important because they increase analyst workload and may lead to alert fatigue. Feature-selection approaches such as CAT-S and collaborative time-series feature selection attempt to remove irrelevant or misleading features. Hybrid models may also target lower false-alarm rates, but a stated design objective is not equivalent to verified low FPR in production. Comparisons should report the actual false-positive rate under explicit evaluation conditions.

Cross Dataset and Temporal Generalization

Cross-dataset studies raise a key limitation of benchmark evaluation: strong performance within a dataset does not establish transfer to another dataset or network environment. Differences in traffic composition, collection period, attack distribution, and feature generation can degrade performance. Random train/test splits from the same benchmark provide weaker evidence of external validity than cross-dataset testing, temporal validation, or evaluation on independently collected traffic. Generalization should therefore be assessed as a distinct dimension rather than inferred from accuracy alone.

Deployment and Implementation Challenges (RQ4)

Four recurring challenges emerge. First, computational cost and resource availability affect deployment in IoT, edge, and industrial-control settings. Lightweight unsupervised methods and tree ensembles may be suitable for constrained hardware, although they can involve trade-offs in detection quality or flexibility. Second, supervised methods depend on labeled data, which can be difficult to obtain for rare or emerging attacks. Federated learning can reduce the need to centralize raw traffic, but communication overhead, client heterogeneity, privacy configuration, and reliance on simulated evaluations remain relevant.

Third, privacy and data-sharing constraints complicate collaborative NIDS development. Federated transfer learning and differential privacy are promising directions, but benchmark results alone do not prove production readiness. Fourth, network traffic and attack behaviour evolve over time, creating a need for monitoring, recalibration, and retraining. Time-series approaches can model temporal behaviour, but changes in the underlying traffic distribution can make previously learned patterns stale.

The practical gap is the difference between laboratory performance and operational effectiveness. Deployment-oriented studies should report latency, memory and compute use, false-positive burden, update requirements, data provenance, and performance on independent traffic. These details are not consistently reported across the selected evidence, limiting conclusions about real-world readiness.

Integrated Discussion

Three cross-cutting conclusions emerge. First, tree ensembles and hybrid deep-learning systems can achieve high benchmark scores, but those figures are conditional on heterogeneous experimental designs and are not directly comparable across papers. Second, dataset and metric choices shape the interpretation of performance: class imbalance, attack-specific outcomes, false positives, and cross-dataset validation deserve explicit treatment. Third, external validity and operational constraints remain major open issues. Lightweight and privacy-preserving approaches address particular deployment needs, but independent, temporal, and cross-environment testing is needed before laboratory performance can be treated as evidence of robust real-world effectiveness.

The contribution of this review is therefore not to name one universally best algorithm. Instead, it links algorithm choice with dataset design, evaluation protocol, and implementation constraints. This integrated view can help researchers design stronger comparisons and help practitioners judge whether published results are relevant to their own networks.

Limitations of the Evidence Base

This draft is grounded in a focused search of accessible publisher and repository pages, not a complete search executed through institutional Scopus, Web of Science, IEEE Xplore, and ACM Digital Library interfaces. The selected studies also span different tasks and environments, and some abstracts do not provide enough detail to compare all experimental procedures. The cross-dataset source by Cantone et al. was retrieved as an arXiv preprint; its peer-reviewed publication status must be confirmed before it is included in a peer-reviewed-only final sample. Before submission, verify all metadata, full-text eligibility, DOI, and publication status, and repeat the database search and PRISMA screening as specified in Methodology.

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

This study reviewed the application of machine learning (ML) techniques for network intrusion detection, emphasizing algorithm performance, dataset selection, evaluation metrics, and implementation challenges. The findings indicate that machine learning provides promising approaches for identifying malicious network traffic and detecting cyberattacks. Ensemble learning methods, including Random Forest and hybrid models, have demonstrated strong performance on several benchmark datasets, while deep learning techniques offer additional capabilities for learning complex traffic patterns and detecting network anomalies.

However, the effectiveness of ML-based intrusion detection systems depends not only on the selected algorithm but also on dataset characteristics, feature engineering, class distribution, and experimental design. High accuracy on public benchmark datasets does not necessarily translate into reliable performance in operational networks. Differences between

training and deployment environments, evolving attack behaviors, and the limited representation of rare attacks remain significant challenges.

The review further highlights the importance of evaluating intrusion detection models using multiple performance indicators. Precision, recall, F1-score, false-positive rate, detection latency, and computational overhead provide a more comprehensive assessment than accuracy alone. Therefore, practical NIDS development requires a balanced consideration of detection effectiveness, computational efficiency, adaptability, and operational reliability.

Overall, no single machine learning method can be considered universally optimal for all network intrusion detection scenarios. The most appropriate approach depends on the target network, threat characteristics, available training data, and deployment constraints. A comprehensive evaluation framework that integrates algorithm performance, dataset suitability, evaluation methodology, and implementation feasibility is essential for advancing reliable and effective ML-based intrusion detection systems.

Recommendations

Based on the findings of this review, several recommendations are proposed for future research and practical implementation.

First, future studies should prioritize cross-dataset and cross-environment evaluation to determine whether models can generalize beyond the datasets used during training. Testing on independent datasets and more realistic network traffic can provide stronger evidence of operational effectiveness.

Second, researchers should address class imbalance and rare attack detection through carefully evaluated techniques such as resampling, cost-sensitive learning, feature selection, and hybrid learning approaches. Performance should be reported separately for individual attack categories where possible, rather than relying exclusively on aggregate accuracy.

Third, studies should adopt a standardized and transparent evaluation protocol. Researchers should clearly document dataset versions, preprocessing procedures, feature selection, training and testing partitions, hyperparameter settings, and evaluation metrics. Such reporting would improve reproducibility and enable more meaningful comparisons between competing methods.

Fourth, future research should investigate computational efficiency and deployment feasibility, particularly for resource-constrained environments, Internet of Things networks, cloud infrastructures, and high-volume enterprise networks. Model complexity, inference latency, memory consumption, and update requirements should be considered alongside detection performance.

Fifth, greater attention should be given to false-positive reduction and model adaptability. Excessive false alarms can increase the workload of security analysts and reduce trust in automated detection systems. Future models should therefore be assessed under changing traffic conditions and evolving attack patterns, including their ability to detect previously unseen threats.

Finally, future systematic reviews should use clearly documented search strategies, explicit inclusion and exclusion criteria, duplicate removal, and a reproducible study-selection process consistent with PRISMA 2020 reporting guidance. The final evidence base should consist of bibliographically verified studies, and quantitative comparisons should only be conducted when the reported results are sufficiently comparable.

In conclusion, advancing machine learning for network intrusion detection requires moving beyond benchmark accuracy toward reliable generalization, transparent evaluation, efficient implementation, and practical cybersecurity value. Addressing these priorities can help bridge the gap between promising experimental results and dependable intrusion detection in real-world network environments.

REFERENCES

- Abbas, Q. (2023). Optimization of predictive performance of intrusion detection system using hybrid ensemble model for secure systems. *PeerJ Computer Science*, 9, e1552. <https://doi.org/10.7717/peerj-cs.1552>
- Abdelmoumin, G., Bensaou, B., & Dustdar, S. (2021). On the performance of machine learning models for anomaly-based intelligent intrusion detection systems for the Internet of Things. *IEEE Internet of Things Journal*. <https://ieeexplore.ieee.org/abstract/document/9509761/>
- Abdullahi, M., Baashar, Y., Al Wahab, N. A., Al-Haiqi, A., & Jamil, M. M. A. (2022). Detecting cybersecurity attacks in Internet of Things using artificial intelligence methods: A systematic literature review. *Electronics*, 11(2), 198. <https://doi.org/10.3390/electronics11020198>
- Al-Ghuwairi, A.-R., et al. (2023). Intrusion detection in cloud computing based on time series anomalies utilizing machine learning. *Journal of Cloud Computing*, 12, 127. <https://doi.org/10.1186/s13677-023-00491-x>
- Alars, Z., & Kurnaz, S. (2024). Enhancing network intrusion detection systems with combined network and host traffic features using deep learning: Deep learning and IoT perspective. *Discover Computing*, 27, 39. <https://doi.org/10.1007/s10791-024-09480-3>
- Ayantayo, A. A., et al. (2023). Network intrusion detection using feature fusion with deep learning. *Journal of Big Data*, 10, 167. <https://doi.org/10.1186/s40537-023-00834-0>
- Harahsheh, A., & Chen, W. (2023). A survey of using machine learning in IoT security and the challenges faced by researchers. *Informatica*, 47, 1–54.
- Kilincer, I. F., Ertam, F., & Sengur, A. (2021). Machine learning methods for cyber security intrusion detection: Datasets and comparative study. *Computer Networks*, 188, 107840. <https://doi.org/10.1016/j.comnet.2021.107840>
- Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E. S. (2019). A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Communications Surveys & Tutorials*, 21(1), 686–728. <https://doi.org/10.1109/COMST.2018.2847722>
- Nazir, A., et al. (2023). A novel feature-selection algorithm in IoT networks for intrusion detection. *Sensors*, 23(19), 8153. <https://doi.org/10.3390/s23198153>
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., et al. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, 372, n71. <https://doi.org/10.1136/bmj.n71>
- Pinto, A., et al. (2023). Survey on intrusion detection systems based on machine learning techniques for the protection of critical infrastructure. *Sensors*, 23(5), 2415. <https://doi.org/10.3390/s23052415>
- Ren, Y., et al. (2023). A lightweight unsupervised intrusion detection model based on variational auto-encoder. *Sensors*, 23(20), 8407. <https://doi.org/10.3390/s23208407>
- Sajid, et al. (2024). Enhancing intrusion detection: A hybrid machine and deep learning approach. *Journal of Cloud Computing*, 13, 123. <https://doi.org/10.1186/s13677-024-00685-x>
- Saranya, T., Sridevi, S., Deisy, C., Chung, T. D., & Khan, M. A. (2020). Performance analysis of machine learning algorithms in intrusion detection system: A review. *Procedia Computer Science*, 171, 1251–1260.
- Shaukat, K., et al. (2020). A survey on machine learning techniques for cyber security in the last decade. *IEEE Access*, 8, 222310–222354.
- Sri Abhijit, et al. (2025). Federated transfer learning for rare attack class detection in network intrusion detection systems. *Scientific Reports*, 15, 33797. <https://doi.org/10.1038/s41598-025-02068-x>

- Stewart, D., Kolajo, T., & Daramola, O. (2024). Machine learning for intrusion detection systems: A systematic literature review. In *Proceedings of FTC 2024* (Lecture Notes in Networks and Systems, Vol. 1154, pp. 623–638). Springer. https://doi.org/10.1007/978-3-031-73110-5_42
- Talukder, M. A., et al. (2024). Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. *Journal of Big Data*, 11, 33. <https://doi.org/10.1186/s40537-024-00886-w>