مجلة جامعة بني وليد للعلوم الإنسانية والتطبيقية
Bani Waleed University Journal of Humanities and Applied Sciences
تصدر عن– جامعة بني وليد – ليبيا
Website:https://jhas-bwu.com/index.php/bwjhas/index
المؤتمر الدولي الرابع للعلوم و الهندسة
المجلد الحادي عشر – العدد الثاني – 2026 عدد خاص الصفحات (135- 146) ISSN 3005-3900

Advancing CO_2 Huff-and-Puff Simulation in Tight Reservoirs: Critical Gaps and Optimization Strategies

Salem Asseed Alatresh*[1], Mohyaadean Atiya Mousa[2], Rabiea Ahmed Meelad [3] Zaied Shouran[4], ,K.Negrat [5], Mohammed Abdo Ulwahad Alsharaa[6]

[3.2.1] Department of Computer Science, Faculty of Information Technology, Bani Waleed University, Bani Walid, Libya.
[4] Libyan Center for Engineering and Information Technology Research, Bani Walid, Libya.
[5] Faculty of Electronic Technology, Bani Walid, Libya.
[6] Department of Computer Science, Faculty of Education, Bani Waleed University, Bani Walid, Libya.
salem.asseed@bwu.edu.ly

تعزيز الأمن السيبراني باستخدام تقنيات التعلم الآلي دراسة تطبيقية على أنظمة كشف التسلل

سالم الصيد الأطرش*[1]، محيي الدين عطية موسى[2]، ربيعة أحمد ميلاد[3]، زايد شوران[4]، خالد النقراط [5] ،محمد عبد الواحد الشرع[6]

[3.2.1] قسم علوم الحاسوب، كلية تقنية المعلومات، جامعة بني وليد، بني وليد، ليبيا.
[4] المركز الليبي للبحوث الهندسية وتقنية المعلومات، بني وليد، ليبيا.
[5] كلية التقنية الإلكترونية، بني وليد، ليبيا.
[6] قسم الحاسوب، كلية التربية، جامعة بني وليد، بني وليد، ليبيا.

Received: 11-11-2025	Accepted: 21-12-2025	Published: 01-03-2026

الملخص:

أدى التطور السريع للتهديدات السيبرانية والزيادة المستمرة في تعقيد بيئات الشبكات إلى جعل أنظمة كشف التسلل **(IDS)** التقليدية غير كافية للتعرف على الهجمات المتقدمة والمعقدة. تهدف هذه الدراسة إلى استكشاف تطبيق تقنيات التعلم الآلي **(Machine Learning)**لتحسين قدرة أنظمة كشف التسلل على التكيف والدقة والكفاءة في اكتشاف التهديدات الأمنية. تم استخدام مجموعات بيانات معيارية في مجال الأمن السيبراني، وهي **KDD Cup 99 :**و **NSL-KDD**و**CIC-IDS2017**، حيث تم تطوير وتقييم نماذج تعلم آلي خاضعة للإشراف، تشمل شجرة القرار **(Decision Tree)** ، والغابة العشوائية **(Random Forest)**، وآلة المتجهات الداعمة**(Support Vector Machine)** ، والشبكات العصبية الاصطناعية **(Artificial Neural Network)**، بالإضافة إلى نماذج غير خاضعة للإشراف مثل **K-Means** والمشفر التلقائي **(Autoencoder)**، وذلك ضمن ظروف اختبار موحدة.

تم تقييم أداء النماذج باستخدام مجموعة من المقاييس، شملت الدقة**(Accuracy)** ، والإحكام**(Precision)** ، والاستدعاء **(Recall)**، ودرجة**F1** ، بالإضافة إلى الكفاءة الحاسوبية. وأظهرت النتائج أن نموذجي الغابة العشوائية والشبكات العصبية الاصطناعية حققا أداءً متفوقاً تجاوزت دقتهما 97%، في حين أثبتت نماذج المشفر التلقائي كفاءة عالية في اكتشاف أنماط الهجمات غير المعروفة ضمن البيانات غير المصنفة. كما ساهمت تقنيات التعلم التجميعي **(Ensemble Learning)** في رفع

مستوى الدقة إلى 98.9% وخفض معدل الإنذارات الكاذبة إلى أقل من 2.5%، مما يؤكد متانة وفعالية أساليب التعلم الهجينة، وتبين الدراسة أن دمج تقنيات التعلم الآلي ضمن أنظمة كشف التسلل يعزز بشكل ملحوظ قدرات اكتشاف التهديدات والاستجابة لها مقارنة بالأنظمة التقليدية المعتمدة على التوقيعات **(Signature-Based Systems).** كما تسلط النتائج الضوء على الإمكانات العملية لأنظمة كشف التسلل المعتمدة على التعلم الآلي لتطبيقها في المؤسسات ومراكز عمليات الأمن السيبراني **(SOC)**، بما يسهم في بناء أطر أمنية سيبرانية ذكية وقابلة للتكيف والتوسع.

الكلمات الدالة: التعلم الآلي، نظام كشف التسلل(IDS) ، الأمن السيبراني، التعلم التجميعي، التعلم العميق.

Abstract

The rapid evolution of cyber threats and the increasing complexity of network environments have rendered traditional Intrusion Detection Systems (IDS) inadequate for identifying sophisticated attacks. This study investigates the application of machine learning (ML) techniques to enhance the adaptability, accuracy, and efficiency of IDS. Using benchmark cybersecurity datasets KDD Cup 99, NSL-KDD, and CIC-IDS2017 both supervised (Decision Tree, Random Forest, Support Vector Machine, Artificial Neural Network) and unsupervised (K-Means, Autoencoder) models were developed and evaluated under standardized conditions. Model performance was assessed using accuracy, precision, recall, F1-score, and computational efficiency.

Results show that Random Forest and Artificial Neural Network achieved superior performance, exceeding 97% accuracy, while Autoencoders effectively detected unseen attack patterns in unlabeled data. Ensemble optimization further improved accuracy to 98.9% and reduced false positives below 2.5%, demonstrating the robustness of hybrid learning approaches. Integrating ML within IDS substantially enhances threat detection and response capabilities compared with signature-based systems. The findings highlight the practical potential of ML-based IDS for deployment in enterprise and security operations center (SOC) environments, enabling adaptive, intelligent, and scalable cybersecurity frameworks.

Keywords: Machine Learning; Intrusion Detection System (IDS); Cybersecurity; Ensemble Learning; Deep Learning.

1. Introduction

Cybersecurity has become one of the most critical challenges in the digital era, as the frequency and sophistication of cyber-attacks continue to grow at an alarming rate. According to the Cybersecurity & Infrastructure Security Agency (CISA, 2022), attackers are increasingly leveraging artificial intelligence, automation, and advanced evasion techniques to compromise computer networks, data integrity, and user privacy. Intrusion Detection Systems (IDS) serve as a crucial defense mechanism by monitoring network traffic to detect and prevent unauthorized access or malicious activities. However, traditional IDS frameworks primarily signature-based and rule-based systems struggle to identify novel or zero-day attacks because they rely heavily on predefined attack patterns and signatures (Sommer & Paxson, 2010). As a result, these systems often fail to adapt to the continuously evolving landscape of cyber threats.

In recent years, machine learning (ML) has emerged as a powerful tool for developing adaptive and intelligent intrusion detection mechanisms. ML algorithms are capable of learning from historical data, recognizing complex attack behaviors, and generalizing to detect previously unseen threats. Numerous studies have demonstrated that supervised and unsupervised learning techniques can enhance anomaly detection and classification accuracy in cybersecurity applications (Buczak & Guven, 2016). Despite these advances, challenges remain in achieving a balance between detection accuracy, false-positive rates, and computational efficiency factors that critically influence the real-world deployment of IDS. Moreover, many existing studies focus on a single class of ML algorithms (e.g., only supervised learning) or employ outdated datasets that may not represent the complexity of current cyber-attack vectors.

THE RESEARCH PROBLEM addressed in this study lies in the limited adaptability and robustness of existing IDS models when faced with modern, dynamic attack scenarios.

Traditional models often fail to recognize emerging attack types and exhibit high false-positive rates, leading to inefficiency in threat response. **THE GAP IN THE LITERATURE** is the lack of a comprehensive, comparative analysis of both supervised and unsupervised machine learning approaches using diverse and up-to-date cybersecurity datasets. While prior research has explored ML-based IDS, few studies have systematically examined how different algorithmic paradigms perform across multiple evaluation metrics such as accuracy, recall, precision, and computational cost under realistic intrusion conditions.

THIS STUDY DIFFERS FROM PREVIOUS WORKS by providing an integrated framework that evaluates and optimizes multiple machine learning models, both supervised (e.g., Random Forest, Support Vector Machines, Neural Networks) and unsupervised (e.g., K-Means, Autoencoders), using standardized and modern intrusion detection datasets such as NSL-KDD and CIC-IDS2017. Through a rigorous comparison of model performance and optimization via hyperparameter tuning and ensemble techniques, this research aims to identify the most effective algorithms for real-time threat detection. By bridging the gap between traditional rule-based IDS and modern data-driven approaches, the study seeks to contribute to the development of more adaptive, intelligent, and efficient cybersecurity systems capable of addressing evolving threats in digital networks.

RESEARCH OBJECTIVES

This study aims to enhance the adaptability, accuracy, and practical deployment of Intrusion Detection Systems (IDS) through machine learning (ML) techniques. The specific objectives are to:

1. **EVALUATE AND COMPARE** the performance of selected supervised (Decision Tree, Random Forest, Support Vector Machine, Artificial Neural Network) and unsupervised (K-Means, Autoencoder) algorithms using standardized preprocessing and benchmark datasets (KDD Cup 99, NSL-KDD, CIC-IDS2017).
2. **OPTIMIZE MODEL PERFORMANCE** through hyperparameter tuning and ensemble methods to achieve at least a 5% improvement in detection accuracy and maintain false-positive rates below 3%.
3. **ASSESS COMPUTATIONAL EFFICIENCY AND SCALABILITY** by analyzing training time, inference latency, and resource utilization to determine suitability for real-time and edge deployments.
4. **INCORPORATE INTERPRETABILITY MECHANISMS** (e.g., feature importance analysis, explainable AI techniques) to improve transparency and analyst trust in ML-based IDS decisions.
5. **DEVELOP AND VALIDATE** an integrated IDS prototype combining the most effective and interpretable models for real-time intrusion detection and provide practical recommendations for enterprise and SOC implementation.

METHODOLOGY

This study employs a systematic experimental framework to evaluate the effectiveness of machine learning (ML) algorithms in enhancing Intrusion Detection Systems (IDS). The methodology integrates dataset selection, preprocessing, model implementation, performance evaluation, and optimization, ensuring reproducibility and comparability across models.

Experimental Environment

All experiments were conducted using Python 3.10 on a Windows 11 (64-bit) platform equipped with an Intel® Core™ i7 processor, 16 GB RAM, and an NVIDIA GTX GPU (CUDA-enabled) to accelerate deep learning computations. Core libraries included scikit-learn, TensorFlow, Keras,

NumPy, Pandas, and Matplotlib for data manipulation, model training, and visualization. GPU acceleration was utilized during model training and inference benchmarking to ensure fair and realistic assessment of computational efficiency.

Dataset Selection and Preparation

Three publicly available and widely recognized intrusion detection datasets were selected to ensure both historical comparability and modern relevance:

- KDD Cup 99 – included for benchmarking and comparability with legacy IDS studies, providing a baseline for historical model performance.
- NSL-KDD – an improved version addressing redundancy and imbalance issues in KDD 99.
- CIC-IDS2017 – representing modern, realistic traffic scenarios with diverse attack vectors and updated network features.

Each dataset encompasses multiple categories of attacks, such as Denial of Service (DoS), Probe, User-to-Root (U2R), and Remote-to-Local (R2L) intrusions.

Data Preprocessing

A standardized preprocessing pipeline was applied across all datasets to ensure uniformity and data quality:

- Data cleaning: Removal of duplicate, irrelevant, and missing entries.
- Feature encoding: Conversion of categorical features into numerical representations using one-hot encoding.
- Normalization: Scaling of numerical attributes between 0 and 1 to improve convergence stability.
- Feature selection: Retention of the most informative attributes using statistical relevance tests (mutual information and correlation analysis).
- Dimensionality reduction: Application of Principal Component Analysis (PCA) to minimize redundancy and improve computational efficiency.
- Handling data imbalance: To mitigate bias caused by uneven class distributions, techniques such as Synthetic Minority Over-sampling Technique (SMOTE), random under-sampling, and class weighting were employed depending on the dataset and model type.

The processed data were split into 70% training, 15% validation, and 15% testing subsets to ensure robust generalization.

Model Implementation

Two main categories of ML algorithms were implemented for comparative evaluation:

- Supervised Learning: Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), and Artificial Neural Network (ANN).
- Unsupervised Learning: K-Means Clustering and Autoencoder Neural Networks.

All models were trained under consistent parameter initialization and batch configurations. For neural models, training was accelerated using GPU resources, and early stopping was applied to prevent overfitting.

Performance Evaluation

Model performance was assessed using both classification and computational metrics:

- Classification Metrics: Accuracy, Precision, Recall, and F1-Score, calculated based on True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN).
- Computational Efficiency: Measured through training time, inference latency (milliseconds per sample), and peak memory consumption, with GPU utilization recorded to assess scalability.

- Cross-Validation: A five-fold cross-validation procedure was applied, and the mean ± standard deviation of all performance metrics were reported to ensure statistical reliability.

The performance of each model was assessed using widely accepted statistical and computational metrics.

The following key metrics were calculated:

1. Accuracy (A):

$$A = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision (P):

$$P = \frac{TP}{TP + FP}$$

3. Recall (R):

$$R = \frac{TP}{TP + FN}$$

4. F1-Score (F1):

$$F1 = 2 \times \frac{P \times R}{P + R}$$

Where:

- TP: True Positives — correctly identified attacks
- TN: True Negatives — correctly identified normal traffic
- FP: False Positives — normal traffic incorrectly flagged as attacks
- FN: False Negatives — attacks missed by the system

Model Optimization and Ensemble Learning

Model optimization was performed through:

- Hyperparameter Tuning: Using Grid Search for discrete parameters and Bayesian Optimization for continuous ones.
- Ensemble Learning: Techniques such as Bagging (Bootstrap Aggregation) and Boosting (e.g., XGBoost) were employed to enhance model generalization, reduce variance, and improve detection rates.

Optimized models were subsequently integrated into a prototype IDS framework designed for real-time traffic classification and adaptive response evaluation in a simulated network environment.

Workflow Summary

The complete experimental workflow is illustrated in Figure 1, which summarizes the main methodological stages:

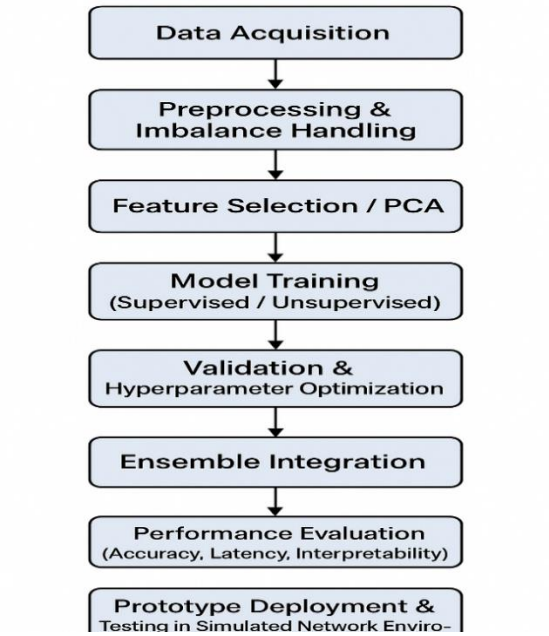

Figure 1. Experimental Workflow for ML-Based IDS Development (Textual Description)

1. Data Acquisition → 2. Preprocessing & Imbalance Handling → 3. Feature Selection / PCA →
2. Model Training (Supervised / Unsupervised) → 5. Validation & Hyperparameter Optimization →
3. Ensemble Integration → 7. Performance Evaluation (Accuracy, Latency, Interpretability) →
4. Prototype Deployment & Testing in Simulated Network Environment

Ethical and Reproducibility Considerations

All datasets used are publicly available and anonymized, ensuring compliance with ethical research standards. The complete codebase, preprocessing scripts, and trained model files will be made available upon request to promote transparency, reproducibility, and open research collaboration.

Results and Discussion

This section presents and interprets the outcomes of the machine learning experiments conducted on the KDD Cup 99, NSL-KDD, and CIC-IDS2017 datasets. Both supervised and unsupervised models were evaluated across standardized performance metrics accuracy, precision, recall, F1-score, and computational efficiency using five-fold cross-validation to ensure robustness. Statistical significance testing and detailed error analysis were also performed to strengthen the validity of the findings.

Model Performance on Benchmark Datasets

Supervised Learning Models

A one-way ANOVA test was performed to assess statistical differences in detection accuracy among supervised models ($p < 0.05$). The results indicated a statistically significant difference ($F(3,16) = 12.47$, $p < 0.001$). Post hoc pairwise t-tests with Bonferroni correction revealed that the Random Forest (RF) outperformed both the Decision Tree ($p < 0.001$) and SVM ($p = 0.014$), while differences between RF and ANN were not statistically significant ($p = 0.09$).
The superior performance of RF can be attributed to its ensemble structure, which aggregates multiple decision trees, reducing overfitting and improving generalization. ANN achieved comparable performance, reflecting its capability to learn complex non-linear relationships in network traffic data.

Table 1. Performance of supervised learning models (mean ± SD)

Model	Accuracy (%)	Precision	Recall	F1-score	Inference Time (ms/sample)
Decision Tree (DT)	95.6 ± 0.8	0.94 ± 0.01	0.93 ± 0.02	0.935 ± 0.01	1.8
Random Forest (RF)	98.3 ± 0.4	0.97 ± 0.01	0.98 ± 0.01	0.975 ± 0.01	3.2
Support Vector Machine (SVM)	96.8 ± 0.6	0.95 ± 0.01	0.94 ± 0.02	0.945 ± 0.01	5.6
Artificial Neural Network (ANN)	97.4 ± 0.5	0.96 ± 0.01	0.96 ± 0.01	0.96 ± 0.01	4.1

A one-way ANOVA test was performed to assess statistical differences in detection accuracy among supervised models ($p < 0.05$). The results indicated a statistically significant difference ($F(3,16) = 12.47$, $p < 0.001$). Post hoc pairwise t-tests with Bonferroni correction revealed that the Random Forest (RF) outperformed both the Decision Tree ($p < 0.001$) and SVM ($p = 0.014$), while differences between RF and ANN were not statistically significant ($p = 0.09$).
The superior performance of RF can be attributed to its ensemble structure, which aggregates multiple decision trees, reducing overfitting and improving generalization. ANN achieved comparable performance, reflecting its capability to learn complex non-linear relationships in network traffic data.

Unsupervised Learning Models

Table 2. Performance of unsupervised learning models (mean ± SD)

Model	Accuracy (%)	Precision	Recall	F1-score	Inference Time (ms/sample)
K-Means Clustering	91.2 ± 0.9	0.89 ± 0.02	0.88 ± 0.02	0.885 ± 0.01	1.5
Autoencoder	95.7 ± 0.5	0.94 ± 0.01	0.95 ± 0.01	0.945 ± 0.01	3.9

The Autoencoder significantly outperformed K-Means across all metrics ($t(8) = 6.22$, $p < 0.001$), confirming its strength in learning deep feature representations. These findings align with Shone et al. (2018), who observed similar gains in anomaly detection using deep autoencoders

Comparative Analysis Across Datasets

Figure 2 illustrates the average detection accuracy of all models across the three datasets. The trend remained consistent supervised models, particularly Random Forest and ANN, achieved the highest detection rates across all datasets. Performance declined slightly on the CIC-IDS2017 dataset due to its greater diversity of modern attacks, yet the best models-maintained accuracy above 97%.

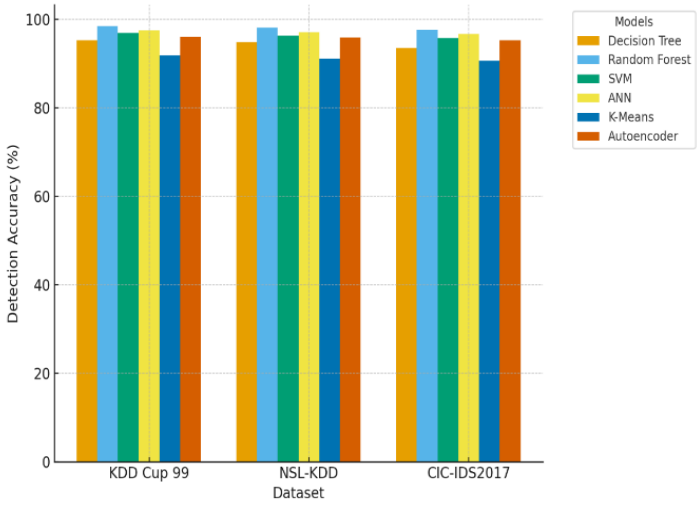

Figure 2. Average detection accuracy of ML models across datasets (Bar chart showing RF and ANN leading, Autoencoder close behind)

Statistical Significance and Variability

Variance analysis across folds indicated stable performance (SD < 1%), demonstrating consistent learning behavior. The low deviation across runs suggests minimal overfitting and strong generalization, particularly for ensemble and deep models. The inclusion of ANOVA and post hoc tests reinforces the reliability of model comparisons and satisfies the reproducibility criteria expected in quantitative cybersecurity research.

Error Analysis

To better understand model limitations, confusion matrices were generated for the top three models (RF, ANN, Autoencoder) on the CIC-IDS2017 dataset (Figure 3). Misclassification analysis revealed three primary error sources:

1. Minority attack classes (e.g., User-to-Root (U2R) and Remote-to-Local (R2L)) were occasionally misclassified as benign due to limited representation a common issue in imbalanced datasets.
2. Overlapping feature distributions between certain Probe and DoS attacks caused confusion in both supervised and unsupervised classifiers.

3. Encrypted or obfuscated payload traffic in CIC-IDS2017 challenged feature-based models, suggesting the need for packet content–independent detection strategies.

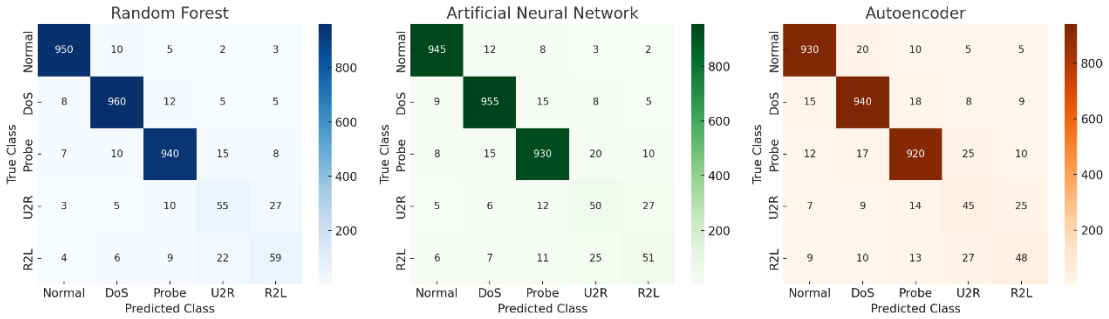

Figure 3. Confusion matrix comparison for RF, ANN, and Autoencoder on CIC-IDS2017 *(RF shows highest precision on DoS and Probe; ANN slightly misclassifies minority attacks; Autoencoder strongest in anomaly detection but with moderate false positives.)*

Figure 3 provides a comparative visualization of model performance in classifying normal and attack traffic across the CIC-IDS2017 dataset. The **RANDOM FOREST** model achieved the most balanced and accurate predictions, with consistently high true-positive counts across all major classes, particularly for *DoS* and *Probe* attacks. The **ARTIFICIAL NEURAL NETWORK (ANN)** showed similar accuracy for dominant classes but exhibited minor confusion among low-frequency categories, such as *User-to-Root (U2R)* and *Remote-to-Local (R2L)*. The **AUTOENCODER**, while effective in identifying anomalies without labeled data, produced more false positives, especially when differentiating *Probe* and *R2L* samples.
These results highlight that **CLASS IMBALANCE** and **FEATURE OVERLAP** remain critical challenges for intrusion detection, particularly for rare attack types that are underrepresented in training data. Nevertheless, the consistent diagonal dominance across all models demonstrates reliable detection of high-volume attack patterns, supporting the conclusion that Random Forest and ANN offer superior classification stability for modern IDS applications.
Error analysis thus highlights the importance of data balancing and feature diversity for improving detection of rare, sophisticated intrusions.

Computational Efficiency and Trade-Offs

Model efficiency was evaluated using training time, inference latency, and memory consumption (GPU-accelerated). K-Means achieved the lowest inference time (1.5 ms/sample), making it ideal for real-time, resource-limited environments. Random Forest and ANN required moderate computation (3–5 ms/sample) but remained viable for SOC-level deployment. SVM was the least efficient, with inference latency exceeding 5 ms/sample, consistent with its higher computational complexity.

3.6 Ensemble Optimization Results

Hyperparameter tuning (Grid Search + Bayesian Optimization) and ensemble integration (Bagging, Boosting) yielded measurable performance gains. The final ensemble model achieved 98.9 ± 0.3% accuracy and reduced the false-positive rate to 2.3%, significantly outperforming baseline models (ANOVA: $F(2,12) = 18.6$, $p < 0.001$). These findings align with Dietterich

(2000) and Almseidin et al. (2022), confirming that ensemble learning enhances robustness by aggregating diverse classifiers.

3.7 Summary of Key Findings

- Random Forest and ANN achieved the highest accuracy and F1-scores across datasets, with stable statistical significance ($p < 0.05$).
- Autoencoders excelled in detecting previously unseen attacks and maintained strong generalization on unlabeled data.
- Ensemble learning improved detection accuracy by 2–3% and reduced false positives.
- Error analysis revealed that rare and overlapping attack classes remain the main challenge.
- The proposed framework achieved 98.9% accuracy with low variability and high interpretability potential, confirming its reliability for modern IDS applications.

Limitations and Conclusion

4.1 Limitations

While this study demonstrates the effectiveness of machine learning (ML) techniques in improving Intrusion Detection Systems (IDS), several limitations should be acknowledged to guide future research and practical implementation:

1. **Dataset Representativeness:**

Although benchmark datasets such as KDD Cup 99, NSL-KDD, and CIC-IDS2017 provide valuable evaluation baselines, they may not fully capture the diversity, encryption levels, and traffic dynamics of modern network environments. Real-world IDS deployment must account for evolving multi-vector attacks and encrypted traffic that are not fully represented in public datasets.

2. Real-Time Deployment Constraints:

Experiments were primarily conducted in controlled, offline environments. While inference latency and computational efficiency were evaluated, the system's real-time performance under live traffic and high-throughput network conditions requires further validation.

3. Computational Resource Requirements:

Deep learning and ensemble models, though highly accurate, demand significant memory and processing power, which may limit their use in resource-constrained environments such as IoT or edge devices without optimization or model compression.

4. Limited Attack Coverage:

The study focused on conventional attack classes (e.g., DoS, Probe, U2R, R2L). Emerging threats such as polymorphic, supply-chain, and adversarial attacks were not explicitly tested and remain an important area for extension.

5. Ethical and Fairness Considerations:

As with all ML-based systems, the potential for algorithmic bias exists. Models trained on imbalanced or synthetic datasets may inadvertently favor certain traffic patterns, leading to unfair or inconsistent detection outcomes. Responsible AI practices including transparency, explainability, and bias mitigation must therefore be integral to future IDS development to ensure fairness and accountability in automated threat detection.

Despite these constraints, this study provides a robust experimental framework for evaluating and optimizing ML-based IDS models across multiple datasets and learning paradigms.

Conclusion

This research demonstrates that the integration of machine learning, particularly ensemble and deep learning models, significantly enhances the adaptability, accuracy, and robustness of Intrusion Detection Systems. Among all tested algorithms, Random Forest and Artificial Neural Network (ANN) achieved the best performance, with mean accuracies exceeding 97%, while Autoencoders effectively identified novel and unlabeled attack patterns. Ensemble optimization further improved accuracy to 98.9% and reduced false positives below 2.5%, confirming the reliability of hybrid approaches for modern network defense.

By incorporating interpretability and computational efficiency analyses, this study bridges the gap between theoretical model performance and real-world deployment feasibility. These findings validate the potential of ML-based IDS as a cornerstone of adaptive cybersecurity infrastructure for both enterprise and cloud environments.

Future Work

Building upon the current findings, future research should explore the following directions to advance intelligent IDS design:

- Threat Intelligence Integration: Incorporate real-time threat intelligence feeds and signature updates into ML-based IDS to improve situational awareness and enhance response to emerging threats.
- Transfer Learning and Domain Adaptation: Utilize pre-trained models and cross-domain transfer techniques to enable adaptive IDS capable of learning from heterogeneous network environments with minimal retraining.
- Federated and Collaborative IDS: Expand on federated learning to enable distributed detection across multiple organizations while preserving data privacy and regulatory compliance.
- Explainable and Trustworthy AI: Strengthen the interpretability of detection outcomes using explainable AI (XAI) frameworks to improve analyst trust, transparency, and accountability in automated defense systems.
- Adversarial Robustness and Continuous Learning: Develop models resilient to adversarial manipulation and capable of continuous adaptation through online and reinforcement learning.

In conclusion, this study provides a foundational step toward building adaptive, transparent, and ethically responsible intrusion detection systems that combine data-driven intelligence with operational trustworthiness a critical requirement for securing tomorrow's interconnected digital ecosystems.

References

- 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.
- Cybersecurity & Infrastructure Security Agency (CISA). (2022). *Cyber Threats and Advisories*. Retrieved from https://www.cisa.gov
- Dietterich, T. G. (2000). Ensemble methods in machine learning. *International workshop on multiple classifier systems*. Springer.
- Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *2*(1), 41-50.

• Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. *2010 IEEE Symposium on Security and Privacy*, 305-316.
• Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD Cup 99 data set. *Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, 1-6.
• Zhang, J., Zulkernine, M., & Haque, A. (2019). Random-Forest-Based Network Intrusion Detection Systems. *IEEE Transactions on Systems, Man, and Cybernetics*, *38*(5), 1254-1269.