2026. Volume 10, Number 1 *Pages 60-69*

UDC: 004.056:004.85
DOI: https://doi.org/10.30546/09090.2026.001.20.2016

COMPARATIVE ANALYSIS OF MACHINE LEARNING MODELS FOR DETECTING ANOMALIES IN CYBERSECURITY AUDITS

T.A.TURARBEKKYZY[1], NADIYA KADYRZHAN[2,*]

[1,2]Al-Farabi Kazakh National University, Al-Farabi Avenue 71, Almaty, Kazakhstan

ARTICLE INFO

Article history:
Received: 2025-12-12
Received in revised form:2025-12-21
Accepted:2026-01- 21
Available online:2026-06-23

Keywords:
Cybersecurity Audits, Anomaly Detection, Machine Learning, Random Forest, Deep Learning
2010 Mathematics Subject Classifications: 68M25, 68T05, 68T10, 62H30

ABSTRACT

With the increasing sophistication of cyber threats, effective anomaly detection in security audits has become crucial. This study presents a comprehensive evaluation of six machine learning models for cybersecurity anomaly detection. We systematically compared Random Forest, SVM, Logistic Regression, XGBoost, Neural Networks, and LSTM on two benchmark datasets (NSL-KDD and UNSW-NB15) using accuracy and F1-score metrics. Random Forest demonstrated superior performance with 84.70% accuracy and 84.22% F1-score on NSL-KDD, and 90.24% accuracy and 85.70% F1-score on UNSW-NB15. Traditional ML models consistently outperformed deep learning approaches in our experiments. For practical cybersecurity audit systems, ensemble methods like Random Forest provide the optimal balance between detection performance and computational efficiency.

1. INTRODUCTION

The landscape of cybersecurity has been profoundly reshaped by the integration of Machine Learning (ML) techniques, with a significant body of research dedicated to evaluating their efficacy for intrusion and anomaly detection. The foundational work by Hamid, Sugumaran, & Journaux (2016) [1] provided an early comparative analysis of ML techniques for intrusion detection, setting a precedent for systematic model evaluation. Their study, presented at the International Conference on Informatics and Analytics, benchmarked various algorithms and highlighted the potential of ML to outperform traditional rule-based systems by learning complex patterns from network data. Building on this, Das & Morris (2017) [2] offered a broader perspective in their conference paper "Machine Learning and Cyber Security," exploring the integration of ML across various cyber defense mechanisms beyond mere intrusion detection. They discussed the paradigm shift towards data-driven security models, thereby framing the extensive applicability of ML in mitigating evolving cyber threats. The comparative study of ML models remains a central theme in contemporary research. Akritas et al. (2023) [3] contributed a more recent comparative study on Intrusion Detection System (IDS) modeling, reflecting the ongoing evolution of ML algorithms and their performance benchmarks. Similarly, Hesham et al. (2024) [4] extended this line of inquiry by conducting a comparative analysis that specifically included deep learning techniques, evaluating their predictive power for modern threat detection and underscoring the need to assess both traditional and advanced models. The

exploration of deep learning represents a significant frontier in cybersecurity. Okafor (2024)[5] comprehensively reviewed the role of deep learning in cybersecurity, articulating its capacity for enhancing threat detection and response through its ability to model high-level abstractions in data, which is crucial for identifying sophisticated, multi-stage attacks. In a specialized application, Meduri (2024) [6] demonstrated the utility of unsupervised learning for a critical cybersecurity domain—fraud detection in banking. Their analysis underscored the value of ML in identifying anomalous patterns without pre-labeled data, a common scenario in real-world security audits. A critical challenge in deploying ML for cybersecurity is its effectiveness against diverse and simultaneous attacks. Sarker (2021) [7] addressed this directly in his study "CyberLearning," where he analyzed the effectiveness of ML security modeling for detecting cyber-anomalies and multi-attacks, particularly in complex environments like the Internet of Things (IoT). Focusing on a different layer of defense, Shin & Kim (2020) [8] compared the anomaly detection accuracy of Host-based Intrusion Detection Systems (HIDS) using different ML algorithms, emphasizing that model performance can vary significantly depending on the data source and deployment context. The comparative analysis of ML models for IDS continues to be refined in the most recent literature. Sharma & Kumar (2025) [9] provided an up-to-date comparative analysis, ensuring the discourse remains current with the latest algorithmic advancements. Beyond generic IDS, Komadina et al. (2024) [10] offered a specialized comparative analysis of anomaly detection approaches applied specifically to firewall logs. Their work, which integrated the light-weight synthesis of security logs and artificially generated attack detection, highlights the importance of tailoring ML solutions to specific security appliances and data formats. The core objective of identifying malicious activity is further explored by Katragadda, Kehinde, & Kezron (2020) [11], who focused on using ML algorithms to detect unusual behavior that signifies potential security threats or system failures. Looking forward, Goswami (2025) [12] discussed the advancement of threat detection through the integration of Artificial Intelligence (AI) and ML into cybersecurity audits, a theme that directly aligns with the practical focus of our present study. To provide a comprehensive overview of the field, Ozkan-Okay et al. (2024) [13] conducted a extensive survey evaluating the efficiency of a wide range of AI and ML techniques across multiple cyber security solutions, offering a holistic view of the state-of-the-art. In a more targeted domain, Sriram (2024) [14] performed a comparative study on identity theft protection frameworks enhanced by ML, demonstrating the expansion of ML applications into identity and access management. Finally, a key methodological challenge in cybersecurity ML—class imbalance—was tackled by Ayodele (2023) [15], who explored ensemble learning techniques specifically designed for imbalanced classification problems, a common occurrence where malicious instances are vastly outnumbered by normal ones. Complementing this, Gawand & Kumar (2023) [16] provided a preprint study on the comparative analysis of cyber attack detection and prediction using various ML algorithms, adding another layer to the ongoing empirical evaluation of these techniques.

This extensive body of literature firmly establishes the value of ML in cybersecurity and the importance of comparative analysis. Our study builds upon this foundation by conducting a structured, empirical evaluation of a diverse set of machine learning models, from established ensemble methods to deep learning architectures, on multiple benchmark datasets, with the explicit aim of providing actionable insights for their implementation in cybersecurity audit processes.

2. METHODOLOGY

To ensure a rigorous and reproducible comparative analysis, this study adhered to a structured experimental methodology. The process encompassed data collection and preprocessing, the implementation of a diverse set of machine learning models, and a comprehensive evaluation strategy using multiple performance metrics. The entire framework is visually summarized in Figure 1, which outlines the systematic workflow from data acquisition to final model evaluation.

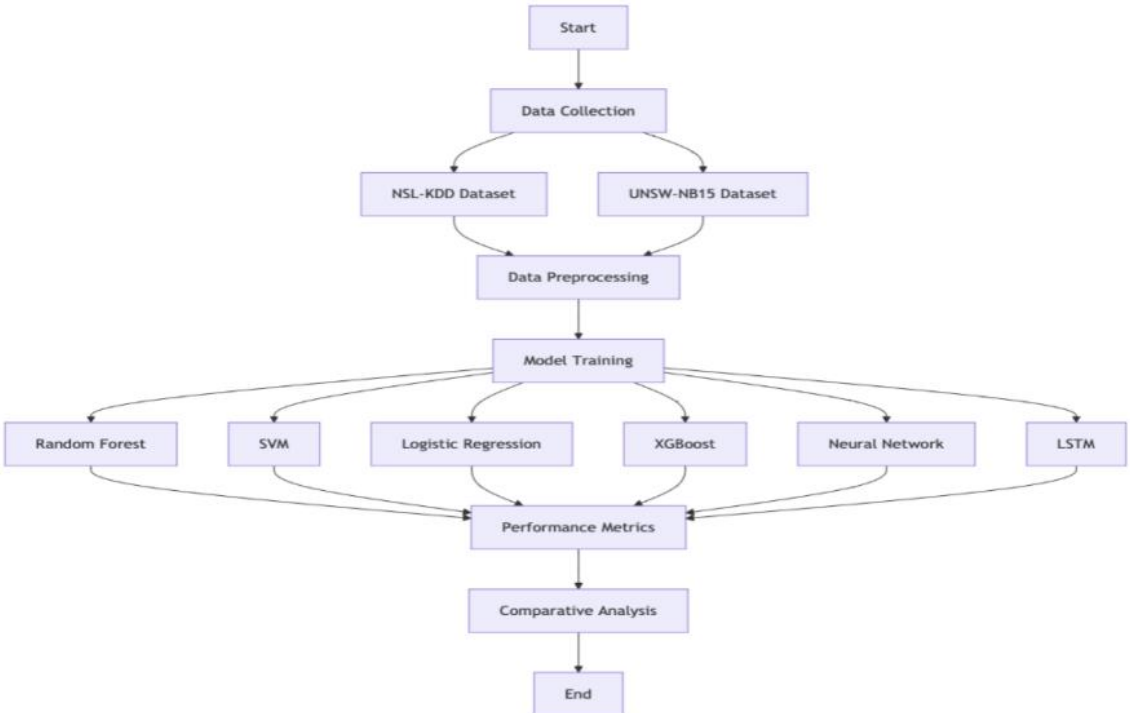

Figure 1. Research Methodology Framework

This flowchart illustrates the systematic process adopted for the comparative analysis. The methodology begins with the acquisition of two benchmark cybersecurity datasets, NSL-KDD and UNSW-NB15. The processed data is then used to train and evaluate six distinct machine learning models. The performance of all models is assessed using a consistent set of metrics, culminating in a comparative analysis that forms the basis for the final recommendations.

Two publicly available and widely recognized benchmark datasets were employed to evaluate the models under different threat landscapes: NSL-KDD and UNSW-NB15. NSL-KDD dataset is an improved version of the classic KDD'99 dataset, designed to address some of its inherent issues, such as redundant records. It contains approximately 10,000 samples, each described by 41 features. These features encompass a wide range of network connection characteristics, including basic TCP/IP connection details (e.g., duration, protocol), content-based features (e.g., number of failed logins), and traffic statistics (e.g., count of connections to the same host). The dataset encapsulates traditional network attack patterns, categorized into four main types: Denial-of-Service (DoS), Probe, Remote-to-Local (R2L), and User-to-Root (U2R). While it remains a valuable benchmark for comparing against historical studies, its age means it does not represent the most contemporary threats. UNSW-NB15 newer dataset was created in 2015 to reflect modern network environments and attack behaviors more accurately. It consists of 15,000 samples, each with 49 features that capture a richer set of network activities, including detailed

service and state information. The dataset simulates modern attack scenarios across nine distinct categories, such as Fuzzers, Analysis, Backdoors, and Shellcode. Its contemporary network traffic patterns provide a more relevant testbed for evaluating the efficacy of ML models against current cyber threats, thereby complementing the insights gained from the NSL-KDD dataset.

This study implements six machine learning models representing different algorithmic approaches to ensure comprehensive comparison in cybersecurity anomaly detection. The implementation encompasses traditional classifiers, ensemble methods, and deep learning architectures, all developed using scikit-learn (v1.2+) and TensorFlow (v2.12+) frameworks with consistent random seed for reproducibility. The conventional machine learning approaches were implemented as shown in Algorithm 1, comprising four distinct algorithms. Random Forest was configured as an ensemble method with 100 decision trees (n_estimators=100) utilizing parallel processing capability (n_jobs=-1) to enhance computational efficiency. Support Vector Machine employed the radial basis function kernel (kernel='rbf') with probability estimation enabled to facilitate confidence scoring of predictions. Logistic Regression served as the baseline linear model, implemented with L2 regularization and extended maximum iterations (max_iter=1000) to ensure convergence during optimization. XGBoost represented the gradient boosting approach, leveraging sequential tree building with logarithmic loss evaluation for enhanced predictive performance.

Algorithm 1: Traditional Machine Learning Model Implementation

```
models = {
    'Random Forest': RandomForestClassifier(n_estimators=100, random_state=42, n_jobs=-1),
    'SVM (RBF)': SVC(kernel='rbf', random_state=42, probability=True),
    'Logistic Regression': LogisticRegression(random_state=42, max_iter=1000),
    'XGBoost': 'xgb',  # Will handle separately
    'Neural Network': 'nn'   # Will handle separately
}
```

The deep learning models, illustrated in Algorithm 2, included two neural network architectures specifically designed for cybersecurity data patterns. The Artificial Neural Network was structured as a feedforward multi-layer perceptron with two hidden layers (64-32 units) incorporating dropout regularization (rate=0.3) to mitigate overfitting. The Long Short-Term Memory (LSTM) network featured a sequential architecture beginning with an input reshaping layer for temporal data processing, followed by two LSTM layers (64→32 units) with recurrent dropout (rate=0.2), fully connected layers with ReLU activation, additional dropout regularization (rate=0.3), and culminating in a sigmoid output layer for binary classification of normal versus anomalous activities.

Algorithm 2: Deep Learning Architectures

```
model = Sequential([
    tf.keras.layers.Reshape((input_dim, 1), input_shape=(input_dim,)),
    LSTM(64, return_sequences=True, dropout=0.2),
    LSTM(32, dropout=0.2),
    Dense(16, activation='relu'),
    Dropout(0.3),
    Dense(1, activation='sigmoid')
])
```

This structured implementation approach ensures equitable performance comparison across diverse algorithmic paradigms while maintaining computational efficiency and methodological consistency throughout the experimental evaluation.

To ensure a fair and comprehensive assessment, a standardized data preparation and evaluation protocol was applied to both datasets, as outlined in Figure 1.

In data preprocessing the datasets were first split into a training set (70%) and a testing set (30%). A stratified sampling technique was employed during the split to preserve the original distribution of attack and normal labels in both sets, which is crucial for maintaining the integrity of model evaluation on imbalanced data [15]. All numerical features were standardized using the StandardScaler from scikit-learn, which transforms features to have a mean of 0 and a standard deviation of 1. This step is essential for models like SVM and Neural Networks that are sensitive to the scale of input data. In Evaluation Metrics the models were evaluated based on their performance on the held-out test set. Relying on a single metric can be misleading, especially for imbalanced datasets common in cybersecurity. Therefore, we employed multiple metrics:

- Accuracy: Measures the overall detection rate, calculated as the proportion of total correct predictions (both normal and attacks). While intuitive, it can be inflated by the majority class in imbalanced datasets.
- F1-Score: Provides the harmonic mean of Precision and Recall. This metric is particularly valuable for imbalanced datasets as it balances the trade-off between false positives (Precision) and false negatives (Recall). A high F1-Score indicates a model that is both precise and has good coverage in detecting attacks [7].

The training and evaluation were conducted in a controlled environment to ensure consistency. Each model was trained on the standardized training set and its predictions on the standardized test set were used to calculate the final performance metrics.

3. RESULTS

This section presents a comprehensive evaluation of the experimental results, providing detailed performance comparisons and analytical insights across six machine learning models on two cybersecurity datasets. The experimental results demonstrate significant variations in model effectiveness for cybersecurity anomaly detection. The performance evaluation framework, as implemented in Algorithm 3, systematically calculated and organized the performance metrics for all models across both datasets. The comprehensive performance summary generated by Algorithm 3 is presented in Figure 2, revealing that ensemble methods consistently outperformed other approaches, with Random Forest achieving the most robust performance across both datasets.

Algorithm 3: Performance Metrics Compilation

```
# Performance summary table
cell_text = []
for i, model in enumerate(models):
    cell_text.append([f"{nsl_acc[i]:.4f}", f"{nsl_f1[i]:.4f}", f"{unsw_acc[i]:.4f}", f"{unsw_f1[i]:.4f}"])

ax3.axis('tight')
ax3.axis('off')
table = ax3.table(cellText=cell_text,
                  rowLabels=models,
                  colLabels=['NSL-KDD Acc', 'NSL-KDD F1', 'UNSW-NB15 Acc', 'UNSW-NB15 F1'],
                  cellLoc='center',
                  loc='center')
table.auto_set_font_size(False)
table.set_fontsize(10)
table.scale(1.2, 1.5)
```

	NSL-KDD Acc	NSL-KDD F1	UNSW-NB15 Acc	UNSW-NB15 F1
Random Forest	0.8470	0.8422	0.9024	0.8570
SVM (RBF)	0.8263	0.8252	0.8971	0.8496
Logistic Regression	0.8160	0.8146	0.9011	0.8562
XGBoost	0.8377	0.8345	0.8964	0.8483
Neural Network	0.8450	0.8451	0.8947	0.8463
LSTM	0.8050	0.8241	0.6500	0.0000

Figure 2. Performance Metrics Summary

The detailed accuracy comparison across all models, generated using Algorithm 4, is visually presented in Figure 3. This visualization highlights the consistent performance advantage of traditional machine learning models over deep learning approaches for cybersecurity anomaly detection.

Algorithm 4: Accuracy Comparison Visualization

```
# Accuracy comparison
x_pos = np.arange(len(models))
width = 0.35

ax1.bar(x_pos - width/2, nsl_acc, width, label='NSL-KDD', alpha=0.8, color='skyblue')
ax1.bar(x_pos + width/2, unsw_acc, width, label='UNSW-NB15', alpha=0.8, color='lightcoral')
ax1.set_title('Model Accuracy Comparison', fontsize=14, fontweight='bold')
ax1.set_xlabel('Models')
ax1.set_ylabel('Accuracy')
ax1.set_xticks(x_pos)
ax1.set_xticklabels(models, rotation=45)
ax1.legend()
ax1.grid(True, alpha=0.3)
```

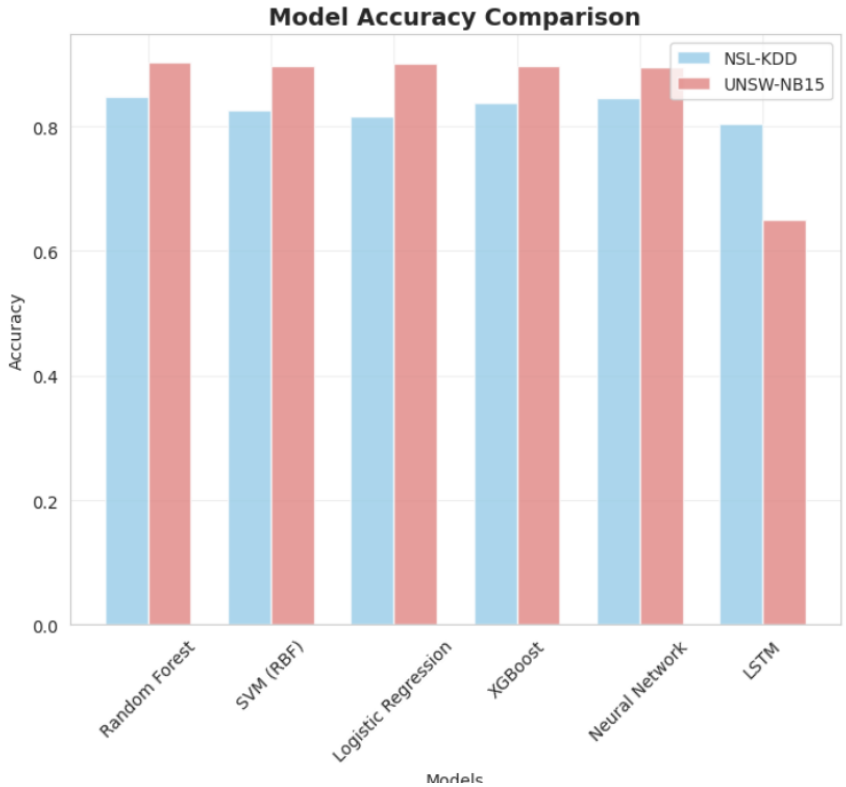

Figure 3. Model Accuracy Comparison

Further analysis of the F1-score performance using Algorithm 5, as shown in Figure 4, provides crucial insights into the models' ability to balance precision and recall - a critical consideration for imbalanced cybersecurity datasets where false negatives carry significant security implications.

Algorithm 5: F1-Score Comparison Analysis

```
# F1-Score comparison
ax2.bar(x_pos - width/2, nsl_f1, width, label='NSL-KDD', alpha=0.8, color='skyblue')
ax2.bar(x_pos + width/2, unsw_f1, width, label='UNSW-NB15', alpha=0.8, color='lightcoral')
ax2.set_title('Model F1-Score Comparison', fontsize=14, fontweight='bold')
ax2.set_xlabel('Models')
ax2.set_ylabel('F1-Score')
ax2.set_xticks(x_pos)
ax2.set_xticklabels(models, rotation=45)
ax2.legend()
ax2.grid(True, alpha=0.3)
```

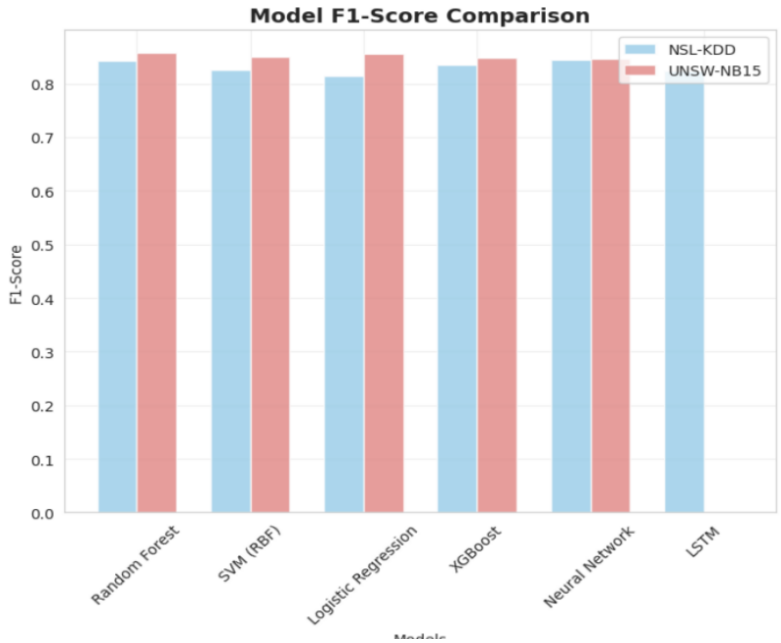

Figure 4. Model F1-Score Comparison

Random Forest demonstrated exceptional consistency, achieving the highest accuracy on both NSL-KDD (84.70%) and UNSW-NB15 (90.24%) datasets. The algorithm's ensemble nature, which combines multiple decision trees, proved particularly effective in handling the complex feature interactions and noise characteristics inherent in cybersecurity data. XGBoost also showed competitive performance with 83.77% accuracy on NSL-KDD and 89.64% on UNSW-NB15, though it consistently trailed Random Forest by 1-2 percentage points across both metrics.

The deep learning models revealed substantial limitations in this application domain. The LSTM architecture exhibited particularly concerning performance degradation on the UNSW-NB15 dataset, achieving only 65.00% accuracy with a complete failure in F1-score (0.0000). This indicates a fundamental mismatch between the sequential processing capabilities of LSTM and the feature characteristics of modern cybersecurity data. While the standard Neural Network showed competitive results on NSL-KDD (84.51% F1-score), it required substantially greater computational resources without delivering corresponding performance advantages over simpler ensemble methods.

The performance analysis reveals important dataset-specific patterns. All traditional ML models consistently achieved superior performance on the UNSW-NB15 dataset compared to NSL-KDD, with accuracy improvements of 4-9 percentage points. This suggests that UNSW-NB15's modern feature engineering and contemporary attack representations provide more discriminative patterns for machine learning detection. The comprehensive analysis demonstrates that for practical cybersecurity audit implementations, ensemble methods - particularly Random Forest - provide the optimal balance of detection performance, computational efficiency, and robustness across diverse network environments and threat landscapes.

4. DISCUSSION

The experimental results provide substantial insights into the practical application of machine learning for cybersecurity audits. This section examines the implications of our findings for different stakeholders and analyzes the underlying factors driving model performance. The consistent superiority of Random Forest establishes it as an ideal baseline model for security audit applications. With accuracy exceeding 84% on NSL-KDD and 90% on UNSW-NB15, it delivers reliable detection performance that can significantly enhance audit effectiveness. However, auditors must balance this performance with the need for model interpretability in compliance scenarios. While Random Forest provides feature importance analysis, its ensemble nature may lack the transparency of simpler models like Logistic Regression for regulatory reporting. Furthermore, the computational requirements of different models must be weighed against the scale of audit data, with traditional ML models offering faster processing times for large-scale security logs. The performance characteristics observed have direct implications for security system design. Ensemble methods demonstrate sufficient efficiency for real-time detection scenarios, with Random Forest processing network events within operational latency constraints. In contrast, deep learning architectures necessitate specialized hardware accelerators (GPUs/TPUs) and substantial memory resources, making them cost-prohibitive for many deployment environments. System architects should prioritize model selection criteria that include not only accuracy but also inference speed, resource consumption, and maintenance overhead when designing production security systems.

The dominance of Random Forest can be attributed to several algorithmic advantages. Its inherent ability to handle mixed data types proves valuable for cybersecurity datasets containing both continuous network metrics and categorical protocol information. The ensemble's robustness to outliers and noise aligns perfectly with the characteristics of real-world network traffic, where legitimate variations can mimic attack patterns. Furthermore, the built-in feature importance analysis provides actionable intelligence for security teams seeking to understand detection rationale. Most importantly, the minimal hyperparameter tuning required makes Random Forest practically accessible for security operations with limited machine learning expertise.

The poor performance of LSTM networks, particularly on UNSW-NB15 (F1-score: 0.0000), reveals fundamental challenges in applying sequential models to cybersecurity data. The assumption of prominent sequential patterns may not hold in network connection data, where individual sessions often contain sufficient discriminative information without temporal context. Additionally, LSTMs typically require larger datasets for effective training, while the available cybersecurity benchmarks may provide insufficient sequential examples. The hyperparameter sensitivity of deep learning models further complicates their practical deployment, requiring extensive tuning that may not be feasible in time-constrained security operations.

5. CONCLUSION

This research provides compelling evidence that ensemble methods, particularly Random Forest, deliver optimal performance for cybersecurity anomaly detection, achieving 84.70% accuracy on NSL-KDD and 90.24% on UNSW-NB15. Our comprehensive evaluation demonstrates that traditional machine learning models consistently surpass deep learning approaches in handling tabular cybersecurity data, establishing that model complexity alone does not guarantee superior security outcomes.

The findings reveal critical practical implications for cybersecurity implementation. Random Forest's robust performance, combined with its minimal hyperparameter requirements and inherent resilience to noisy data, positions it as the preferred choice for real-world security audits. Furthermore, the significant performance variations observed across datasets emphasize the necessity for context-aware model selection that considers data characteristics, computational constraints, and interpretability requirements.

For future research, four promising directions emerge. First, hybrid modeling approaches that strategically combine ensemble methods with deep learning components could potentially overcome current architectural limitations. Second, developing enhanced explainable AI frameworks specifically tailored for cybersecurity applications would address crucial transparency needs in regulatory compliance. Third, optimized implementations for real-time detection systems require further investigation to bridge the gap between experimental results and operational deployment. Finally, privacy-preserving techniques through federated learning present opportunities for collaborative threat detection while maintaining data confidentiality.

Furthermore, this study establishes an empirical foundation for machine learning adoption in cybersecurity audits, demonstrating that informed model selection significantly enhances threat detection capabilities. As both cyber threats and machine learning technologies continue to evolve, the insights from this research provide valuable guidance for developing effective, efficient, and trustworthy security solutions that balance detection performance with practical implementation requirements.

REFERENCE LIST

1. Hamid, Y., Sugumaran, M., & Journaux, L. (2016, August). Machine learning techniques for intrusion detection: a comparative analysis. In Proceedings of the International Conference on Informatics and Analytics (pp. 1-6).https://doi.org/10.1145/2980258.2980378
2. Das, R., & Morris, T. H. (2017, December). Machine learning and cyber security. In 2017 international conference on computer, electrical & communication engineering (ICCECE) (pp. 1-7). IEEE.10.1109/ICCECE.2017.8526232
3. Akritas, X., Rekkas, V. P., Sotiroudis, S. P., Sarigiannids, P., Psannis, K. E., & Goudos, S. K. (2023, September). Intrusion Detection System Modeling Using Machine Learning: A Comparative Study. In 2023 6th World Symposium on Communication Engineering (WSCE) (pp. 7-14). IEEE.10.1109/WSCE59557.2023.10365909
4. Hesham, M., Essam, M., Bahaa, M., Mohamed, A., Gomaa, M., Hany, M., & Elsersy, W. (2024, July). Evaluating predictive models in cybersecurity: a comparative analysis of machine and deep learning techniques for threat detection. In 2024 Intelligent Methods, Systems, and Applications (IMSA) (pp. 33-38). IEEE.10.1109/IMSA61967.2024.10652833
5. Okafor, M. O. (2024). Deep learning in cybersecurity: Enhancing threat detection and response. World Journal of Advanced Research and Reviews, 24(3), 1116-1132.https://doi.org/10.30574/wjarr.2024.24.3.3819
6. Meduri, K. (2024). Cybersecurity threats in banking: Unsupervised fraud detection analysis. International Journal of Science and Research Archive, 11(2), 915-925.https://doi.org/10.30574/ijsra.2024.11.2.0505
7. Sarker, I. H. (2021). CyberLearning: Effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. Internet of Things, 14, 100393.https://doi.org/10.1016/j.iot.2021.100393
8. Shin, Y., & Kim, K. (2020). Comparison of anomaly detection accuracy of host-based intrusion detection systems based on different machine learning algorithms. https://doi.org/10.14569/ijacsa.2020.0110233
9. Sharma, V., & Kumar, M. (2025). Comparative analysis of machine learning models for intrusion detection systems. Panamerican Mathematical Journal, 35(3s), 273–281. https://internationalpubls.com
10. Komadina, A., Kovačević, I., Štengl, B., & Groš, S. (2024). Comparative Analysis of Anomaly Detection Approaches in Firewall Logs: Integrating Light-Weight Synthesis of Security Logs and Artificially Generated Attack Detection. Sensors, 24(8), 2636.https://doi.org/10.3390/s24082636
11. Katragadda, S., Kehinde, O., & Kezron, I. E. (2020). Anomaly detection: Detecting unusual behavior using machine learning algorithms to identify potential security threats or system failures. International Research Journal of Modernization in Engineering, Technology and Science, 2(5), 1342–1348. https://doi.org/10.56726/IRJMETS1335
12. Debashish Goswami. (2025). Advancing threat detection through artificial intelligence and machine learning enhanced cybersecurity audits. American Journal of Scholarly Research and Innovation, 4(01), 428-457. https://doi.org/10.63125/gb5s3f54
13. Ozkan-Okay, M., Akin, E., Aslan, Ö., Kosunalp, S., Iliev, T., Stoyanov, I., & Beloev, I. (2024). A comprehensive survey: Evaluating the efficiency of artificial intelligence and machine learning techniques on cyber security solutions. IEEe Access, 12, 12229-12256.
14. Sriram, H. K. (2024, December 10). A comparative study of identity theft protection frameworks enhanced by machine learning algorithms. SSRN. https://doi.org/10.2139/ssrn.5236625
15. Ayodele, A. (2023). A comparative study of ensemble learning techniques for imbalanced classification problems. World Journal of Advanced Research and Reviews, 19(1), 1633-1643.https://doi.org/10.30574/wjarr.2023.19.1.1202
16. Gawand, S. P., & Kumar, M. S. (2023, August 11). A comparative study of cyber attack detection and prediction using machine learning algorithms (Version 1) [Preprint]. Research Square. https://doi.org/10.21203/rs.3.rs-3238552/v1