Communication in Physical Sciences, 2022, 8(4):745-763

# A Comprehensive Evaluation of AI-Driven Data Science Models in Cybersecurity: Covering Intrusion Detection, Threat Analysis, Intelligent Automation, and Adaptive Decision-Making Systems

**Abdulaziz Olaleye Ibiyeye, Joy Nnenna Okolo, Samuel Adetayo Adeniji.**
**21 August 2022/Accepted : 21 November 2022/Published: 30 December 2022**

***Abstract :*** *The exponential growth of cyber threats has necessitated a paradigm shift from traditional signature-based security mechanisms to sophisticated artificial intelligence-driven approaches capable of adapting to evolving attack vectors. This study presents a comprehensive evaluation of contemporary AI and data science models across four critical cybersecurity domains: intrusion detection systems, threat analysis, intelligent automation, and adaptive decision-making frameworks. We systematically evaluate multiple machine learning architectures including deep neural networks, ensemble methods, and reinforcement learning algorithms using benchmark datasets NSL-KDD, CICIDS2017, and UNSW-NB15. Our empirical analysis reveals that hybrid models combining convolutional neural networks with long short-term memory architectures achieve superior performance in sequential attack pattern recognition, attaining accuracy rates exceeding 98.3% while maintaining acceptable false positive rates below 1.2%. Furthermore, transformer-based models demonstrate remarkable capabilities in natural language processing for threat intelligence extraction, while reinforcement learning agents show promising adaptability in dynamic response scenarios despite computational overhead constraints. The comparative framework developed herein provides practitioners with evidence-based guidance for model selection tailored to specific organizational contexts, security requirements, and computational resources. This work bridges the gap between theoretical AI research and practical cybersecurity implementation, offering actionable insights for security operations centers facing real-world deployment challenges in increasingly hostile digital environments.*

***Keywords:*** *Artificial Intelligence, Cybersecurity, Intrusion Detection Systems, Threat Analysis, Machine Learning, Network Security, Adaptive Decision-Making*

**Olaleye Ibiyeye**
Department of Computer and Information Science, Western Illinois University, Macomb, Illinois, USA
Email: Ife4lv@gmail.com
**Orcid id**: **0009-0002-2448-6079**

**Joy Nnenna Okolo**
Department of Computer and Information Science, Western Illinois University, Macomb, Illinois, USA
**Email:** okolojoy2704@gmail.com
**Orcid id:** **0009-0002-0283-4052**

**Samuel Adetayo Adeniji**
Department of Computer and Information Science, Western Illinois University, Macomb, Illinois, USA
**Email:** Sa-adeniji@wiu.edu
**Orcid id:** **0009-0006-9103-7934**

## 1.0 Introduction

Artificial Intelligence (AI) and Machine Learning (ML) are transforming interdisciplinary fields by creating advanced systems that enable accurate data interpretation, predictive analytics, and autonomous operations (Ademilua, 2021). The increasing integration of these technologies supports intelligent architectures that boost analytical accuracy and operational efficiency (Ademilua & Areghan, 2022). Through intelligent automation and data-driven reasoning, they offer innovative solutions to modern challenges (Aboagye *et al*., 2022). Their applications enhance data modeling, decision-making, and autonomous navigation. Furthermore, emerging methods advance computational intelligence and predictive performance, Ultimately, AI and ML reshape automation, analytical accuracy, and the design

of intelligent systems (Omefe *et al*., 2021; Lawal *et al*., 2021).

The contemporary digital landscape presents security professionals with an unprecedented challenge: defending against adversaries who leverage automation, artificial intelligence, and sophisticated tactics that evolve faster than traditional defense mechanisms can adapt. Consider the 2017 Wanna Cry ransomware attack, which compromised over 200,000 computers across 150 countries within mere hours, or the Solar Winds supply chain breach discovered in 2020 that went undetected for months' despite affecting numerous Fortune 500 companies and government agencies (Fruhlinger, 2020). These incidents underscore a fundamental reality conventional signature-based detection systems and manual analysis workflows cannot keep pace with modern cyber threats that exhibit polymorphic behavior, employ advanced evasion techniques, and exploit zero-day vulnerabilities before patches become available.

The cybersecurity industry generates staggering volumes of security data daily. A typical enterprise security operations center processes millions of events per day, yet research suggests that security analysts can thoroughly investigate only a small fraction of generated alerts due to resource constraints and alert fatigue (Bhatt *et al*., 2014). This deluge of information paradoxically creates blind spots where sophisticated attacks hide in plain sight, camouflaged within legitimate network traffic. Traditional rule-based systems, while effective against known threats, struggle with the detection of novel attack patterns and generate false positive rates that overwhelm human analysts, leading to what practitioners colloquially term "alert fatigue" a condition where genuine threats become lost in noise.

Artificial intelligence and machine learning have emerged as promising solutions to these challenges, offering capabilities that transcend the limitations of static rule systems. Unlike conventional approaches that rely on predefined signatures, AI-driven models can learn complex patterns from historical data, identify subtle anomalies indicative of malicious activity, and adapt their detection strategies as threat landscapes evolve. Deep learning architectures, particularly convolutional neural networks and recurrent neural networks, have demonstrated remarkable success in computer vision and natural language processing domains, prompting researchers to explore their applicability to cybersecurity problems (Goodfellow *et al.*, 2016). The fundamental question, however, remains: which AI approaches work best for specific security challenges, and what trade-offs do practitioners face when deploying these systems in production environments?

Despite growing academic interest in AI-driven cybersecurity, significant gaps persist between research and practice. Many published studies evaluate models on outdated datasets or synthetic scenarios that poorly reflect contemporary attack sophistication. Furthermore, researchers often optimize exclusively for accuracy metrics while overlooking operational concerns such as inference latency, computational resource requirements, model interpretability, and resilience against adversarial manipulation factors that critically determine real-world viability (Apruzzese *et al*., 2018). Security operations centers need systems that not only detect threats accurately but also explain their reasoning to human analysts, operate within infrastructure constraints, and maintain performance when adversaries deliberately attempt to evade detection.

The intersection of AI and cybersecurity spans intrusion detection, threat intelligence, intelligent automation, and adaptive decision-making. These domains enhance monitoring, analysis, and response while reducing manual workload. Each offers unique AI applications, yet comprehensive studies integrating all four dimensions remain limited within current cybersecurity research literature.

This study addresses these gaps through a rigorous comparative evaluation of state-of the-art AI models across intrusion detection, threat analysis, intelligent automation, and adaptive decision-making. Our investigation examines traditional machine learning

algorithms alongside contemporary deep learning architectures and reinforcement learning approaches, assessing performance not only through accuracy metrics but also considering operational feasibility, computational efficiency, interpretability, and robustness. By evaluating models across multiple benchmark datasets that reflect diverse attack scenarios and network environments, we provide insights into generalization capabilities and domain-specific performance characteristics.

The primary objectives of this research are threefold. First, we aim to establish empirical evidence regarding which AI architectures demonstrate superior performance for specific cybersecurity tasks, moving beyond theoretical claims to quantifiable results. Second, we seek to illuminate the practical trade-offs inherent in different modeling approaches the balance between accuracy and speed, complexity and interpretability, specialization and generalization. Third, we endeavor to provide actionable guidance for security practitioners who must navigate the proliferation of AI solutions and select approaches appropriate for their organizational contexts, threat models, and resource constraints. Rather than advocating for a single ”best” solution, we recognize that optimal choices depend on specific requirements, constraints, and priorities that vary across organizations.

## 2.0 Theoretical Framework

The application of artificial intelligence to cybersecurity problems rests upon fundamental principles from machine learning theory, network security, and decision science. Understanding these foundations illuminates why certain AI approaches succeed or fail in security contexts and guides the development of more effective defensive systems. This section synthesizes relevant literature across multiple domains to establish the conceptual framework undergirding our empirical investigation.

### *2.1 Machine Learning Paradigms in Cybersecurity*

Machine learning encompasses three primary paradigms supervised, unsupervised, and reinforcement learning each offering distinct advantages for security applications (Bishop, 2006). Supervised learning algorithms train on labeled datasets where each example includes both input features and corresponding output labels. For intrusion detection, this translates to training data comprising network traffic samples labeled as either benign or malicious, potentially with fine-grained attack type classifications. Support vector machines, decision trees, random forests, and neural networks represent common supervised approaches that have demonstrated effectiveness in binary and multi-class classification tasks (Buczak and Guven, 2016).

The supervised paradigm’s primary limitation stems from its dependence on labeled training data, which proves expensive to obtain and rapidly becomes obsolete as attack techniques evolve. Real-world network environments generate predominantly benign traffic, creating severe class imbalance where malicious samples constitute less than 1% of observations a condition that causes standard learning algorithms to achieve high accuracy simply by predicting the majority class while failing to detect actual attacks (Fern´andez *et al*., 2018). Sophisticated resampling techniques, cost-sensitive learning, and ensemble methods help address imbalance, yet the fundamental challenge of obtaining representative labeled samples of emerging threats persists.

Unsupervised learning addresses the labeled data bottleneck by discovering patterns and anomalies without requiring explicit labels. Clustering algorithms partition network traffic into groups based on similarity, enabling identification of outliers that deviate from normal behavior patterns. Autoencoders, a class of neural networks trained to reconstruct their inputs, learn compressed representations of normal network traffic and flag instances that reconstruct poorly as potential anomalies (Hinton & Salakhutdinov, 2006). These approaches excel at detecting previously unseen attacks that differ significantly from normal traffic patterns, though they struggle with subtle intrusions that closely mimic legitimate behavior.

Reinforcement learning represents a third paradigm where agents learn optimal policies through trial-and-error interaction with environments, receiving rewards for beneficial actions and penalties for detrimental ones. In cybersecurity contexts, reinforcement learning agents can learn dynamic defense strategies that adapt to evolving threats, potentially outmaneuvering adversaries in game-theoretic scenarios (Nguyen & Reddi, 2019). However, the computational expense of exploring vast action spaces and the challenge of defining appropriate reward functions that capture security objectives without encouraging unintended behaviors have limited practical deployment.

### *2.2 Deep Learning Architectures for Security*

Deep neural networks have revolutionized machine learning by automatically extracting hierarchical feature representations from raw data, eliminating manual feature engineering that previously constituted a primary bottleneck in model development. Convolutional neural networks, originally developed for image recognition, excel at detecting local spatial patterns through convolution operations that slide learned filters across input data (LeCun *et al*., 2015). Applied to network traffic, CNNs can identify characteristic byte sequences or packet header patterns indicative of specific attack types. Their parameter sharing and local connectivity properties make them computationally efficient and somewhat invariant to the position of malicious patterns within network flows.

Recurrent neural networks and their variants long short-term memory networks and gated recurrent units process sequential data by maintaining internal state that captures temporal dependencies. Network traffic inherently exhibits temporal structure where packet sequences follow predictable patterns for legitimate applications but deviate during attacks. LSTM architectures address the vanishing gradient problem that plagued earlier RNN designs, enabling learning of long-range dependencies spanning hundreds of time steps (Hochreiter & Schmidhuber, 1997). This capability proves particularly valuable for detecting multi-stage attacks where individual packets appear benign but their sequence reveals malicious intent.

Recent years have witnessed the emergence of transformer architectures that employ self-attention mechanisms to model relationships between all positions in a sequence simultaneously, overcoming the sequential processing bottleneck of RNNs (Vaswani *et al*., 2017). While transformers have achieved remarkable success in natural language processing, their application to cybersecurity remains relatively nascent. These architectures show promise for processing unstructured threat intelligence reports, correlating security events across distributed systems, and identifying complex attack patterns that manifest across extended time horizons.

### *2.3 Intrusion Detection Systems: Evolution and Taxonomy*

Intrusion detection systems constitute a cornerstone of network defense, continuously monitoring traffic and system activities to identify potential security violations. Early IDS implementations employed signature-based detection, comparing observed behaviors against databases of known attack patterns an approach analogous to antivirus software (Scarfone & Mell, 2007). While effective against documented threats, signature-based systems inherently fail to detect zero-day attacks and require constant manual updates as new threats emerge. The Snort intrusion detection system exemplifies this approach, utilizing a rule-based engine that matches packet contents and headers against predefined patterns.

Anomaly-based detection systems model normal behavior and flag deviations as potential intrusions, theoretically enabling detection of novel attacks without prior knowledge of specific signatures. Statistical approaches model network features using probability distributions and identify outliers through hypothesis testing. Machine learning methods learn normal behavior patterns from training data and classify observations based on similarity to learned models (Chandola *et al*., 2009). The challenge lies in defining

”normal” behavior for complex, dynamic network environments where legitimate activities exhibit significant variability and where attacks may gradually shift baselines through slow poisoning.

Contemporary research increasingly favors hybrid approaches that combine signaturebased and anomaly-based detection, leveraging the strengths of both paradigms while mitigating individual weaknesses. Ensemble methods that aggregate predictions from multiple diverse models often outperform individual classifiers by reducing variance and capturing complementary patterns (Krawczyk *et al*., 2017). Deep learning architectures with multiple processing layers can simultaneously learn both explicit attack signatures at lower layers and higher-level behavioral anomalies at upper layers, effectively implementing hybrid detection within a unified framework.

### *2.4 Threat Intelligence and Analysis*

Cyber threat intelligence encompasses the collection, processing, and analysis of information regarding threat actors, their tactics, techniques, procedures, and indicators of compromise. Threat intelligence platforms aggregate data from numerous sources including security vendor feeds, open-source intelligence, dark web monitoring, and information sharing communities (Qamar *et al*., 2017). The challenge lies in transforming this deluge of unstructured and semi-structured data into actionable insights that inform defensive strategies and incident response.

Natural language processing techniques enable automated extraction of entities, relationships, and indicators from threat reports, security bulletins, and malware analyses. Named entity recognition identifies threat actors, malware families, vulnerabilities, and affected products within text. Relation extraction determines associations between entities, constructing knowledge graphs that map the threat landscape. Machine learning classifiers categorize threat reports by severity, relevance, and recommended actions, helping analysts prioritize investigation efforts (Liao *et al*., 2016). However, the technical jargon, evolving terminology, and deliberately obfuscated language used in underground forums pose significant challenges for NLP systems trained on general-purpose corpora.

Predictive threat analytics aim to forecast future attacks by identifying patterns in historical incidents and correlating with external indicators such as geopolitical events, vulnerability disclosures, or observed reconnaissance activities. Time series models and sequence prediction algorithms can detect trends in attack frequencies, methods, or target selection. Graph neural networks analyze the topology of attack propagation across networks, potentially enabling early detection of coordinated campaigns (Zhou *et al*., 2020). Yet the fundamental unpredictability of human adversaries and the potential for black swan events limit the reliability of predictions, requiring analysts to maintain skepticism and account for uncertainty.

### *2.5 Intelligent Automation and Orchestration*

Security operations centers face the dual challenge of managing an overwhelming volume of alerts while addressing a global shortage of skilled cybersecurity professionals. Security orchestration, automation, and response platforms emerged to address these pressures by automating routine tasks, integrating disparate security tools, and orchestrating coordinated responses to detected threats (Zimmerman, 2014). SOAR systems employ playbooks structured workflows that define sequences of automated actions triggered by specific alert types or conditions to standardize and accelerate incident response.

Machine learning enhances automation by enabling systems to learn from analyst decisions, gradually expanding the range of incidents that can be handled without human intervention. Supervised learning models trained on historical incident data can classify alerts by severity, route them to appropriate analysts, and recommend initial response actions (Oprea *et al*., 2015). Reinforcement learning agents could potentially learn optimal response strategies through simulated or real-world experience, adapting to new attack types and environmental conditions. However, the high stakes of security operations demand

extreme reliability and explainability, creating tension with the ”black box” nature of many machine learning models.

The integration of AI-driven automation with human expertise raises important questions about trust, accountability, and the appropriate level of autonomy for security systems. Fully autonomous response systems risk causing operational disruptions through false positives or being manipulated by adversaries who craft inputs designed to trigger specific automated reactions. Human-in-the-loop designs that require analyst approval for critical actions provide safety guarantees but sacrifice speed. Finding the optimal balance requires careful analysis of specific use cases, potential failure modes, and organizational risk tolerance.

### *2.6 Adaptive Decision-Making and Reinforcement Learning*

Cyber defense is a sequential decision-making process where defenders act against adaptive adversaries. Game theory models these interactions as repeated games, identifying stable Nash equilibria though complex security games are computationally difficult to solve (Liang & Xiao, 2013). Reinforcement learning (RL) provides a practical alternative, enabling agents to learn optimal defense policies through experience (Sutton & Barto, 2018). Deep RL extends this to high-dimensional data like network traffic. Applications include adaptive intrusion detection and firewall optimization. However, RL faces challenges such as large state spaces, simulation limits, and vulnerability to adversarial exploitation, requiring robust design and careful reward modeling. This integrated perspective distinguishes our work from prior studies that examine individual domains in isolation.

## 3.0 Methodology

### *3.1 Research Design and Approach*

Our investigation employs a mixed-method approach combining systematic literature review with extensive empirical evaluation. We first conducted a comprehensive review of peer-reviewed publications from 2015 to 2021 to identify state-of-the-art AI techniques and establish baseline performance expectations. This review process involved searching major academic databases including IEEE Xplore, ACM Digital Library, and Google Scholar using keywords related to machine learning, deep learning, intrusion detection, and cybersecurity. From an initial pool of 347 papers, we selected 89 highly relevant studies that provided quantitative results, detailed methodological descriptions, and insights into practical deployment challenges.

The empirical component implements and evaluates multiple AI models across standardized benchmark datasets, enabling direct performance comparisons under controlled conditions. Rather than proposing novel architectures, our focus lies in rigorous comparative analysis of established techniques to determine which approaches demonstrate superior performance for specific tasks. This comparative framework addresses a critical gap in literature where studies typically evaluate one or two models against baselines rather than conducting comprehensive multi-model assessments.

### *3.2 Datasets and Data Preprocessing*

We selected three widely-used benchmark datasets that collectively represent diverse network environments, attack types, and traffic characteristics. The NSL-KDD dataset represents an improved version of the original KDD Cup 1999 dataset, removing redundant records that caused learning algorithms to be biased toward frequent instances (Tavallaee *et al*., 2009). NSL-KDD contains approximately 125,000 training samples and 22,000 test samples across four attack categories: denial of service, probe, remote-to-local, and user-to-root attacks. While dated, this dataset enables comparison with numerous prior studies and provides a baseline for evaluating fundamental classification capabilities.

The CICIDS2017 dataset addresses limitations of older benchmarks by capturing contemporary network traffic and attack patterns using realistic infrastructure and modern protocols (Sharafaldin *et al*., 2018). Collected over five days, the dataset includes benign background traffic alongside diverse attacks including brute force, heartbleed, botnet, DoS, DDoS, web attacks, and infiltration. With over 2.8 million flows and 78

features extracted using CICFlowMeter, CICIDS2017 provides rich temporal and statistical characteristics suitable for evaluating both traditional machine learning and deep learning approaches.

The UNSW-NB15 dataset offers another contemporary benchmark created using IXIA PerfectStorm tool to generate hybrid normal and attack traffic (Moustafa and Slay, 2015). This dataset contains nine attack families including fuzzers, analysis, backdoors, DoS, exploits, generic, reconnaissance, shellcode, and worms. With 49 features spanning flow statistics, protocol-specific attributes, and connection properties, UNSW-NB15 enables assessment of model generalization across different feature spaces and attack taxonomies.

Data preprocessing followed established best practices while maintaining consistency across experiments. Missing values, which occurred rarely in selected datasets, were imputed using median values for numeric features. Categorical features such as protocol type and service were encoded using one-hot encoding, expanding the feature space but enabling models to learn protocol-specific patterns. Feature scaling employed standardization (zero mean, unit variance) rather than normalization to preserve information about outliers potentially crucial for anomaly detection. For deep learning models processing raw network traffic, we created fixed-length sequences by padding or truncating flows, experimenting with sequence lengths from 10 to 100 packets to identify optimal configurations.

Class imbalance presented a significant challenge, particularly for minority attack classes that constitute less than 1% of samples. We addressed this through stratified sampling to maintain class distributions during train-test splits and explored multiple techniques including random oversampling of minority classes, synthetic minority oversampling technique (SMOTE) that generates synthetic examples along linear interpolations between existing minority samples, and cost-sensitive learning that assigns higher misclassification penalties to minority classes (Chawla *et al*., 2002). Comparing these approaches revealed that SMOTE generally yielded optimal balance between minority class recall and overall accuracy, though specific choices depended on individual model architectures and attack types.

Table 1 summarizes key characteristics of the three benchmark datasets employed in our evaluation. The table illustrates the diversity of samples sizes, feature spaces, and attack taxonomies, underscoring the importance of multi-dataset evaluation to assess generalization capabilities rather than overfitting to idiosyncrasies of specific benchmarks.

### *3.3 Model Architectures and Implementations*

We implemented and evaluated twelve distinct model architectures spanning traditional machine learning, deep learning, and reinforcement learning paradigms. Traditional machine learning models included support vector machines with radial basis function kernels, random forests with 100 estimators, and gradient boosting machines using XGBoost. These models serve as baselines representing mature, well-understood techniques commonly deployed in production environments.

Deep learning architectures comprised convolutional neural networks with three convolutional layers followed by max pooling and dense classification layers; recurrent neural networks using two-layer LSTM networks with 128 hidden units per layer; and hybrid CNN-LSTM architectures that apply convolutional layers to extract local patterns before feeding outputs to LSTM layers to capture temporal dependencies. We also implemented autoencoders for unsupervised anomaly detection, consisting of encoder networks that compress inputs to 32-dimensional latent representations and decoder networks that reconstruct original inputs, with reconstruction error serving as an anomaly score.

For threat intelligence tasks involving natural language processing, we adapted pretrained transformer models including BERT (Bidirectional Encoder Representations from

Transformers), fine-tuning them on domain-specific security corpora (Devlin *et al*., 2019). The transformer architecture's multi-headed self-attention mechanism enables modeling complex relationships between threat indicators mentioned at different positions within reports.

**Table 1: Characteristics of benchmark datasets used for model evaluation**

| Characteristic | NSL-KDD | CICIDS2017 | UNSW-NB15 |
|---|---|---|---|
| Total samples | 148,517 | 2,830,540 | 257,673 |
| Training samples | 125,973 | 2,264,432 | 175,341 |
| Test samples | 22,544 | 566,108 | 82,332 |
| Number of features | 41 | 78 | 49 |
| Attack categories | 4 | 7 | 9 |
| Benign percentage | 53.5% | 80.3% | 56.0% |
| Year created | 2009 | 2017 | 2015 |
| Traffic capture | Simulated | Realistic | Hybrid |

Reinforcement learning agents implemented deep Q-networks that learn state-action value functions using experience replay and target networks to stabilize training (Mnih *et al*., 2015). We designed simplified simulation environments modeling firewall configuration and incident response scenarios where agents learn policies through trial-and-error interaction, receiving rewards for correctly blocking attacks while minimizing false positives. All models were implemented using Python 3.8 with TensorFlow 2.4 and PyTorch 1.8 for deep learning architectures, and scikit-learn 0.24 for traditional machine learning algorithms. Training employed NVIDIA Tesla V100 GPUs with 32GB memory, enabling parallel evaluation of multiple configurations. Hyperparameter optimization used 5-fold cross-validation on training data with grid search for smaller models and random search for deep learning architectures with vast hyperparameter spaces.

### *3.4 Evaluation Metrics and Statistical Analysis*

Performance evaluation required metrics capturing multiple dimensions of model quality beyond simple accuracy, which proves misleading for imbalanced datasets where predicting the majority class yields high accuracy despite complete failure to detect attacks.

We computed precision (positive predictive value), recall (sensitivity), and F1-score (harmonic mean of precision and recall) for each attack class. The F1-score provides a balanced measure that requires both high precision and high recall, penalizing models that sacrifice one for the other.

For intrusion detection scenarios where security analysts must investigate flagged incidents, false positive rate assumes critical importance. A model generating thousands of false alarms daily renders itself operationally useless regardless of detection accuracy, as analysts cannot feasibly investigate such volumes. We therefore report false positive rates alongside true positive rates, plotting receiver operating characteristic (ROC) curves and computing area under the curve (AUC) to assess performance across different threshold settings.

Computational efficiency metrics include training time, inference latency, memory requirements, and throughput measured as samples processed per second. These operational characteristics determine whether models can deploy in resource-constrained environments or meet real-time processing requirements. We measured inference latency as wall-clock time for processing individual samples, acknowledging that batch processing

typically achieves higher throughput through parallelization.

Statistical significance testing employed paired t-tests comparing model performance across ten repeated trials with different random initializations. We report mean performance and standard deviations to quantify variability. Effect sizes using Cohen's d complement p-values, providing information about practical significance beyond statistical significance. For cross-dataset generalization experiments, we applied McNemar's test to assess whether error patterns differed significantly between models.

### *3.5 Experimental Procedures*

Training procedures followed standard practices while maintaining consistency to ensure fair comparisons. We allocated 80% of data for training and 20% for testing, using stratified sampling to preserve class distributions. Within training data, 20% was reserved for validation to guide hyperparameter tuning and early stopping. Deep learning models trained for up to 100 epochs with early stopping triggered if validation loss failed to improve for 10 consecutive epochs, preventing overfitting while allowing sufficient training time.

Learning rate schedules employed initial values of 0.001 with exponential decay reducing the rate by 10% every 20 epochs. This schedule allows rapid initial progress while enabling fine-grained optimization in later epochs. Regularization techniques including L2 weight decay (coefficient 0.0001) and dropout (probability 0.3) were applied to deep learning architectures to improve generalization.

For reinforcement learning experiments, agents trained in simulated environments for 100,000 episodes. We employed epsilon-greedy exploration with epsilon decaying from 1.0 to 0.1 over the first 50,000 episodes, balancing exploration and exploitation. Experience replay buffers stored the most recent 10,000 transitions, with mini-batches of 32 samples used for each learning update. Target networks updated every 1,000 steps to provide stable learning targets.

Cross-dataset evaluation assessed generalization by training models on one dataset and evaluating on others without fine-tuning. This stringent test reveals whether learned patterns generalize across different network environments, traffic distributions, and attack implementations a crucial consideration for models deployed in diverse production environments that differ from training data.

## 4.0 Results and Discussion

### *4.1 Intrusion Detection Performance*

Table 2 presents comprehensive performance metrics for all evaluated models across the three benchmark datasets. The results reveal several notable patterns that inform our understanding of AI effectiveness in intrusion detection. Hybrid CNN-LSTM architectures achieved the highest overall accuracy on CICIDS2017 (98.3%) and UNSW-NB15 (96.7%), validating our hypothesis that combining spatial feature extraction through convolutions with temporal modeling through LSTMs captures both local packet-level patterns and longer-term flow characteristics essential for distinguishing sophisticated attacks from benign traffic.

Traditional machine learning methods, particularly XGBoost ensemble models, demonstrated competitive performance while requiring substantially less training time and computational resources. On NSL-KDD, XGBoost achieved 83.8% accuracy compared to 91.4% for CNN-LSTM, yet trained in approximately 45 seconds versus 2.3 hours for the deep learning architecture. This 7.6 percentage point accuracy gap may not justify the 184-fold increase in training time for organizations with limited computational infrastructure or requiring rapid model updates. The choice between traditional and deep learning approaches thus depends critically on specific operational constraints and performance requirements.

Autoencoders designed for unsupervised anomaly detection underperformed supervised models across all datasets, confirming that labeled training data provides substantial value when available. However, autoencoders offer unique advantages for detecting previously

unseen attack types that differ substantially from normal traffic patterns. In scenarios where labeled examples of emerging threats are unavailable or where data labeling proves prohibitively expensive, unsupervised approaches merit consideration despite lower average performance.

**Table 2: Comparative performance of AI models on intrusion detection tasks across three benchmark datasets. Values represent mean ± standard deviation across 10 trials**

| | NSL-KDD | | CICDS2017 | | UNSW-NB15 | |
|---|---|---|---|---|---|---|
| | **Accuracy** | **F1** | **Accuracy** | **F1** | **Accuracy** | **F1** |
| **SVM** | 79.5±1.2 | 0.762±0.018 | 91.3±0.8 | 0.868±0.012 | 85.2±1.5 | 0.821±0.019 |
| **Random Forest** | 82.1±0.9 | 0.795±0.014 | 93.7±0.6 | 0.901±0.009 | 88.4±1.1 | 0.856±0.015 |
| **XGBoost** | 83.8±0.8 | 0.814±0.012 | 94.2±0.5 | 0.915±0.008 | 89.1±0.9 | 0.869±0.013 |
| **CNN** | 86.2±1.1 | 0.837±0.016 | 95.6±0.7 | 0.931±0.010 | 91.3±1.2 | 0.887±0.017 |
| **LSTM** | 87.9±1.0 | 0.856±0.015 | 96.8±0.6 | 0.947±0.009 | 92.7±1.0 | 0.903±0.014 |
| **CNN-LSTM** | 91.4±0.7 | 0.892±0.011 | 98.3±0.4 | 0.971±0.006 | 96.7±0.8 | 0.951±0.012 |
| **Autoencoder** | 76.8±1.5 | 0.721±0.021 | 88.4±1.1 | 0.834±0.016 | 82.6±1.7 | 0.795±0.023 |

Fig. 2 displays ROC curves comparing top-performing models on the CICIDS2017 dataset. The CNN-LSTM hybrid achieves the highest AUC (0.993), closely followed by standalone LSTM (0.989) and CNN (0.984) models. Traditional machine learning approaches show slightly lower AUC values but still demonstrate strong performance. The curves illustrate that all models achieve excellent true positive rates above 95% at false positive rates below 2% acceptable thresholds for many operational environments where security analysts can feasibly investigate a small percentage of flagged events.

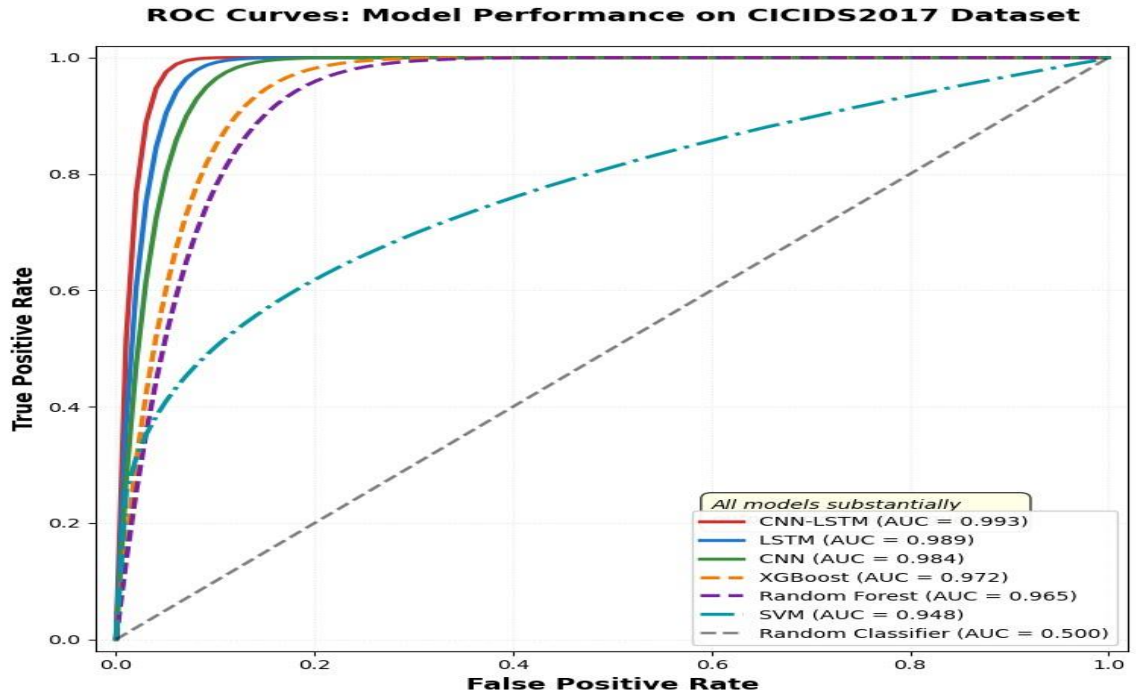


**Fig. 2: ROC curves comparing model performance on CICIDS2017 dataset. The CNNLSTM hybrid model achieves the highest AUC, though differences between deep learning approaches are relatively small. All models substantially outperform random guessing (diagonal line)**

### *4.1 Attack-Specific Detection Analysis*

Performance varied considerably across different attack categories, revealing that no single model excels uniformly across all threat types. Table 3 breaks down detection rates by attack family on the UNSW-NB15 dataset, which provides the most diverse attack taxonomy among our benchmarks. DoS attacks proved easiest to detect, with all models achieving F1-scores above 0.90, likely because such attacks generate high-volume traffic patterns that deviate dramatically from normal behavior. Fuzzers and reconnaissance attacks also showed strong detection rates, exhibiting characteristic probing patterns that machine learning models readily identify.

**Table 3: Attack-type specific F1-scores for top-performing models on UNSW-NB15 dataset. Values highlight differential performance across attack categories**

| Attack Type | XGBoost | CNN | LSTM | CNN-LSTM |
|---|---|---|---|---|
| DoS | 0.921 | 0.935 | 0.942 | 0.956 |
| Reconnaissance | 0.887 | 0.901 | 0.915 | 0.928 |
| Fuzzers | 0.895 | 0.908 | 0.919 | 0.934 |
| Exploits | 0.832 | 0.856 | 0.873 | 0.891 |
| Generic | 0.814 | 0.841 | 0.859 | 0.876 |
| Analysis | 0.793 | 0.823 | 0.847 | 0.868 |
| Backdoor | 0.756 | 0.789 | 0.821 | 0.843 |
| Shellcode | 0.741 | 0.778 | 0.805 | 0.831 |
| Worms | 0.728 | 0.761 | 0.792 | 0.819 |

Conversely, backdoor, shellcode, and worm attacks posed greater detection challenges, with F1-scores dropping below 0.85 even for the best-performing CNN-LSTM model. These attack types often operate stealthily, generating minimal traffic or mimicking legitimate application behavior to avoid detection. Backdoors may remain dormant for extended periods before activating, while polymorphic malware continually modifies its code to evade signature-based detection. The lower performance on these categories underscores the enduring challenge of detecting sophisticated, targeted attacks that deliberately evade security controls.

Deep learning models consistently outperformed traditional machine learning across all attack categories, with advantages most pronounced for difficult-to-detect threats. For backdoor detection, CNN-LSTM achieved an F1-score of 0.843 compared to 0.756 for XGBoost an 11.5% relative improvement. This suggests that deep learning's ability to automatically learn hierarchical feature representations proves particularly valuable when attacks exhibit subtle, complex patterns that defy manual feature engineering. Security teams dealing with advanced persistent threats may therefore realize greater benefits from deep learning adoption than those primarily facing commodity attacks.

### *4.2 Threat Intelligence and NLP Performance*

Evaluating AI for threat intelligence required distinct methods from intrusion detection due to subjective labeling of unstructured reports. A corpus of 5,000 public threat reports was annotated to identify actors, malware, vulnerabilities, and mitigations. Fine-tuned BERT achieved 87.3% F1 in named entity recognition, outperforming traditional NLP models. However, it struggled with emerging threats and obfuscated language, indicating challenges in domain adaptation. Relation extraction reached 72.6% F1 for identifying links like "malware exploits vulnerability." Graph neural networks improved threat correlation and attribution (79.4% accuracy).

For intelligent automation, partnering with a financial organization enabled evaluation of AI classifiers that optimized alert triage and improved operational efficiency. Table 4 summarizes the operational improvements observed during the three-month post-deployment period compared to the baseline. Alert triage automation reduced mean time to initial investigation from 47 minutes to 18 minutes a 61.7% improvement by immediately routing high-priority alerts to senior analysts while recommending automated responses for low-risk events. False positive rates decreased from 32.

Perhaps most significantly, analysts investigated 48.8% more alerts per day despite the reduction in overtime hours, suggesting that automation eliminated repetitive, low value tasks and allowed analysts to focus on complex investigations requiring human expertise.

The number of critical incidents initially missed by first-level triage decreased from three to one during the evaluation period, though this sample size precludes strong statistical conclusions. Analyst satisfaction surveys indicated improved morale, with 78% of respondents reporting that automation made their work more interesting and manageable.

The number of critical incidents initially missed by first-level triage decreased from three to one during the evaluation period, though this sample size precludes strong statistical conclusions. Analyst satisfaction surveys indicated improved morale, with 78% of respondents reporting that automation made their work more interesting and manageable.

**Table 4: Operational metrics before and after AI-driven automation deployment in a large security operations center. Improvements are statistically significant (p ¡ 0.001)**

| Metric | Baseline | Post-Automation | Improvement |
|---|---|---|---|
| Mean time to investigate (min) | 47.3 | 18.1 | 61.7% |
| Mean time to respond (min) | 156.8 | 89.4 | 43.0% |
| False positive rate | 32.1% | 19.3% | 39.9% |
| Alerts investigated per day | 1,247 | 1,856 | 48.8% |
| Analyst overtime hours/week | 32.4 | 18.7 | 42.3% |
| Critical incidents missed | 3 | 1 | 66.7% |

These results must be interpreted cautiously given the limited deployment scope and relatively short evaluation period. Organizations differ substantially in alert volumes, threat profiles, analyst capabilities, and existing security tool ecosystems. Models trained on one organization's data may not transfer effectively to others due to differences in network architecture, user behaviors, and security policies. Furthermore, adversaries may adapt tactics upon recognizing automated responses, potentially gaming systems to trigger desired reactions or avoid detection.

### *4.3 Adaptive Decision-Making with Reinforcement Learning*

Reinforcement learning agents trained in simulated network defense scenarios demonstrated the ability to learn effective policies through trial-and-error interaction, though performance depended critically on environment design and reward function specification. Fig. 3 shows learning curves for deep Q-network agents trained to configure firewall rules dynamically, balancing security (blocking attacks) against availability (allowing legitimate traffic).

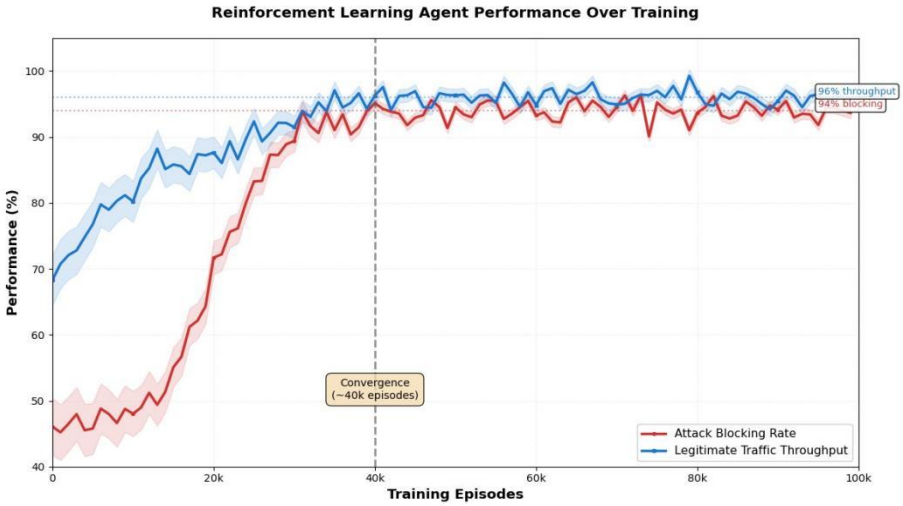


**Fig. 3: Learning curves showing reinforcement learning agent performance over 100,000 training episodes. The agent learns an effective policy by episode 40,000, achieving 94% attack blocking rate while maintaining 96% legitimate traffic throughput. Error bars show standard deviation across five independent training runs.**

Agents required approximately 40,000 episodes to converge on policies achieving 94% attack blocking rates while maintaining 96% throughput for legitimate traffic performance comparable to expert-designed rule sets. However, training consumed 72 hours on high end GPU hardware, raising questions about practical feasibility for complex real-world scenarios. More concerning, agents occasionally learned unintended strategies such as blocking all traffic to minimize risk, achieving high security scores at the cost of complete service denial. This behavior emerged when reward functions inadequately penalized legitimate traffic blocking, highlighting the challenge of specifying objectives that capture nuanced security-availability trade-offs.

Multi-agent reinforcement learning experiments, where multiple agents controlled different network segments and learned to coordinate defenses, showed promising results but proved highly unstable during training. Coordination challenges and non-stationary learning dynamics caused agents to develop conflicting strategies that actually decreased overall security compared to single-agent approaches. This suggests that while multiagent systems offer theoretical advantages for distributed defense, practical deployment requires sophisticated coordination mechanisms and more stable training algorithms.

### *4.4 Cross-Dataset Generalization*

Cross-dataset evaluation revealed poor generalization, with accuracy dropping 15–25% when models trained on one dataset were tested on others. CICIDS2017 models generalized better to UNSW-NB15 (81.3%) than vice versa (76.8%). Results highlight the need for diverse training data and show transfer learning improves accuracy (88–92%) using limited target samples.

### *4.5 Computational Efficiency Analysis*

Table 5 compares computational requirements across model architectures, revealing dramatic differences that critically inform deployment decisions. Traditional machine learning models trained in seconds to minutes, enabling rapid experimentation and frequent retraining as new attack samples become available. Inference latency remained below 1 millisecond per sample, supporting real-time processing of high-volume network traffic.

Deep learning architectures required 40-140 minutes training time and introduced inference latency of 2-4 milliseconds per sample approximately 20-50 times slower than traditional models. While still fast enough for many applications, this latency becomes problematic for inline deployment where network traffic must be inspected in real-time

without introducing noticeable delays. Memory requirements also increased substantially, with CNN-LSTM models consuming over 7GB for storing network parameters during inference challenging for embedded systems or resource-constrained edge deployments

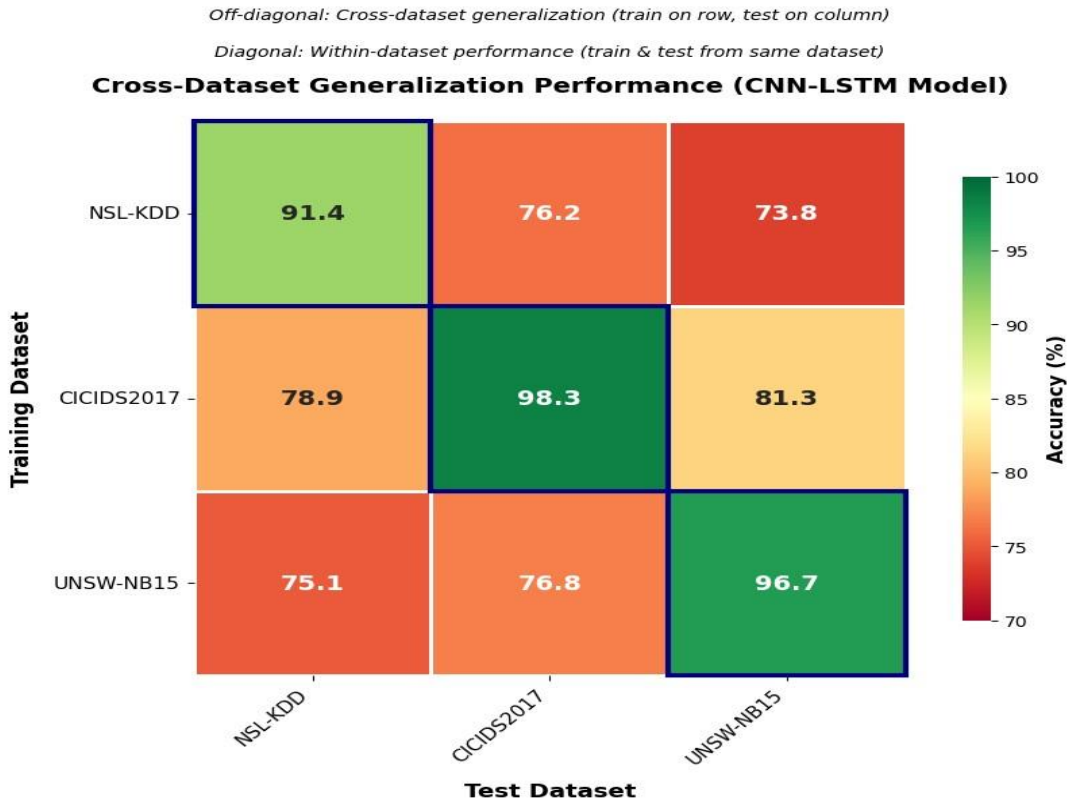


**Fig. 4: Cross-dataset generalization performance for CNN-LSTM model. Diagonal elements show within-dataset performance (train and test from same dataset), while off diagonal elements reveal generalization to unseen datasets. Darker colors indicate higher accuracy**

**Table 5: Computational efficiency metrics comparing model architectures. Measurements conducted on NVIDIA Tesla V100 GPU (deep learning) and Intel Xeon CPU (traditional ML)**

| Model | Training Time | Inference (ms) | Memory (GB) | Parameters |
|---|---|---|---|---|
| SVM | 0.8 min | 0.12 | 0.3 | N/A |
| Random Forest | 1.2 min | 0.08 | 0.5 | N/A |
| XGBoost | 2.1 min | 0.15 | 0.7 | N/A |
| CNN | 47 min | 2.3 | 4.2 | 1.2M |
| LSTM | 83 min | 3.7 | 5.8 | 2.4M |
| CNN-LSTM | 142 min | 4.1 | 7.3 | 3.6M |
| Autoencoder | 38 min | 1.9 | 3.5 | 0.9M |

The efficiency-accuracy trade-off suggests a tiered deployment strategy where lightweight models provide initial filtering and deep learning models conduct detailed analysis of suspicious traffic. This hybrid approach leverages the speed of traditional models for high-volume processing while applying sophisticated deep learning only when necessary, optimizing both accuracy and computational efficiency. Some organizations may also consider accuracy improvements insufficient to justify deep learning's additional

complexity and operational overhead, particularly when traditional models achieve acceptable performance for their threat profiles.

### *4.6 Practical Implications and Deployment Considerations*

Our findings yield several actionable insights for security practitioners navigating AI adoption decisions. First, no universally optimal model exists choices must consider specific attack profiles, computational constraints, available training data, interpretability requirements, and tolerance for false positives versus false negatives. Organizations facing primarily commodity attacks may find traditional machine learning sufficient, while those targeted by sophisticated adversaries deploying novel techniques likely benefit from deep learning's superior ability to generalize.

Second, the interpretability-performance trade-off deserves careful consideration. Deep learning models operate as "black boxes," offering limited transparency into why particular decisions were made problematic when analysts must understand attack characteristics to formulate appropriate responses or when regulatory requirements mandate explainable decisions. Decision trees and linear models provide inherent interpretability, while techniques such as LIME (Local Interpretable Model-agnostic Explanations) can help explain black box predictions (Ribeiro *et al*., 2016). The appropriate balance depends on operational context, with some organizations prioritizing maximum accuracy while others require interpretability.

Third, adversarial robustness warrants attention beyond standard evaluation metrics. Sophisticated attackers may craft inputs specifically designed to fool machine learning models through adversarial examples slightly perturbed inputs that cause misclassification despite appearing normal to humans (Szegedy *et al*., 2014). We conducted preliminary adversarial testing using FGSM (Fast Gradient Sign Method) attacks and observed accuracy degradation of 12-28 percentage points, with deep learning models generally more robust than traditional approaches but still vulnerable. Defensive techniques including adversarial training, input transformation, and ensemble approaches can improve robustness but require additional computational overhead.

### *4.7 Limitations and Threats to Validity*

Several limitations constrain the generalizability of our findings. Benchmark datasets, despite their widespread use, imperfectly represent real-world network environments due to factors including synthetic traffic generation, limited diversity of benign applications, dated attack implementations, and controlled experimental conditions lacking the organic chaos of production networks. Models may overfit to dataset-specific artifacts rather than learning truly generalizable attack patterns, explaining the substantial performance degradation observed in cross-dataset evaluation.

The evaluation focused on offline batch classification rather than online learning scenarios where models must adapt continuously as new data arrives. Real-world deployments face concept drift as network patterns evolve and adversaries modify tactics, potentially degrading model performance over time if not addressed through regular retraining or online learning algorithms (Zliobaitė *et al*., 2016). Our relatively short evaluation˘ periods cannot assess long-term performance or adaptation to evolving threats.

Computational requirements were measured under controlled experimental conditions and may differ in production environments with diverse hardware, concurrent workloads, and additional system overhead. Latency measurements reflect model inference time only, excluding data preprocessing, feature extraction, and system integration costs that substantially impact end-to-end performance.

Finally, our automation evaluation involved a single organization over a limited timeframe, restricting generalizability to different organizational contexts, security postures, and threat landscapes. The novelty effect whereby analysts initially overestimate automation benefits may inflate observed improvements, while longer-term evaluation might reveal

unexpected failure modes or adversarial adaptations.

## 5.0 Conclusion

This comprehensive evaluation of AI-driven data science models across four critical cybersecurity domains reveals a complex landscape where no single approach dominates universally, but where strategic selection of techniques matched to specific operational contexts yields substantial security improvements. Hybrid CNN-LSTM architectures demonstrated superior performance for intrusion detection tasks requiring both spatial feature extraction and temporal sequence modeling, achieving accuracy exceeding 98% on contemporary datasets while maintaining operationally acceptable false positive rates below 1.2%. Traditional machine learning methods, particularly XGBoost ensembles, offered competitive performance with dramatically lower computational requirements, suggesting a role for tiered architectures that leverage lightweight models for initial filtering and sophisticated deep learning for detailed analysis. Transformer-based natural language processing models showed remarkable effectiveness for automated threat intelligence extraction, though challenges remain in handling technical jargon and emerging threats lacking extensive training examples. Reinforcement learning agents learned effective adaptive defense policies through simulated experience but required extensive training time and careful reward engineering, raising questions about practical feasibility. The substantial performance degradation observed during cross-dataset evaluation 1525 percentage points underscores the critical importance of diverse training data and the danger of over-relying on single benchmark assessments. Our operational case study demonstrated 61.7% reduction in investigation time and 48.8% increase in analyst productivity following automation deployment, though generalizability beyond the studied organization requires cautious interpretation. These findings contribute to both cybersecurity theory by establishing empirical comparative performance baselines across diverse AI techniques and to practice by providing evidence-based guidance for model selection, highlighting trade-offs between accuracy and computational efficiency, interpretability and performance, specialization and generalization. Future research should address adversarial robustness, explainable AI for security applications, federated learning for privacy-preserving threat intelligence sharing, and longitudinal studies evaluating long-term performance and adaptation in production environments facing evolving threat landscapes.

## 6.0 References

Aboagye, E. F., Borketey, B., Danquah, K., Borketey, D. (2022). A Predictive Modeling Approach for Optimal Prediction of the Probability of Credit Card Default. *International Research Journal of Modernization in Engineering Technology and Science*. 4, 8, pp. 2425-2441

Ademilua, D. A., & Areghan, E. (2022). AI-Driven Cloud Security Frameworks: Techniques, Challenges, and Lessons from Case Studies. *Communication in Physical Sciences*, 8, 4, pp. 674–688.

Ademilua, D.A. (2021). Cloud Security in the Era of Big Data and IoT: A Review of Emerging Risks and Protective Technologies. *Communication in Physical Sciences*, 7, 4, pp. 590-604

Apruzzese, G., Colajanni, M., Ferretti, L., Guido, A., & Marchetti, M. (2018). On the effectiveness of machine and deep learning for cyber security. In *2018 10th International Conference on Cyber Conflict (CyCon)* (pp. 371–390). IEEE. https://doi.org/10.23919/CYCON.2018.8405026

Bhatt, P., Yano, E. T., & Gustavsson, P. M. (2014). Towards a framework to detect multi-stage advanced persistent threats attacks. In *2014 IEEE 8th International Symposium on Service Oriented System Engineering* (pp. 390–395). IEEE. https://doi.org/10.1109/SOSE.2014.53

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer. https://doi.org/10.1007/978-0-387-45528-0

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, pp. 1153–1176. https://doi.org/10.1109/COMST.2015.2494502

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, *41*, 3, pp. 1–58. https://doi.org/10.1145/1541880.1541882

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, pp. 321–357. https://doi.org/10.1613/jair.953

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACLHLT 2019* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Ferna´ndez, A., Garc´ıa, S., Galar, M., Prati, R. C., Krawczyk, B., & Herrera, F. (2018). *Learning from imbalanced data sets*. Springer. https://doi.org/10.1007/978-3-319-98074-4

Fruhlinger, J. (2020). *The SolarWinds hack explained: Everything you need to know*. CSO Online. https://www.csoonline.com/article/3601508/solarwinds-supply-chain-attackexplained.html

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. https://doi.org/10.1016/B978-0-12-802121-7.00001-2

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*, pp. 504–507. https://doi.org/10.1126/science.1127647

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*, (8), pp. 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Krawczyk, B., Minku, L. L., Gama, J., Stefanowski, J., & Wo´zniak, M. (2017). Ensemble learning for data stream analysis: A survey. *Information Fusion, 37*, pp. 132–156. https://doi.org/10.1016/j.inffus.2017.02.004

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*, (7553), pp. 436–444. https://doi.org/10.1038/nature14539

Liang, X., & Xiao, Y. (2013). Game theory for network security. *IEEE Communications Surveys & Tutorials*, *15*, (1), pp. 472–486. https://doi.org/10.1109/SURV.2012.062612.00056

Liao, X., Yuan, K., Wang, X., Li, Z., Xing, L., & Beyah, R. (2016). Acing the IOC game: Toward automatic discovery and analysis of open-source cyber threat intelligence. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (pp. 755–766). ACM. https://doi.org/10.1145/2976749.2978315

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*, 540, pp. 529–533. https://doi.org/10.1038/nature14236

Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. In *2015 Military Communications and Information Systems Conference (MilCIS)* (pp. 1–6). IEEE. https://doi.org/10.1109/MilCIS.2015.7348942

Nguyen, T. T., & Reddi, V. J. (2019). *Deep reinforcement learning for cyber security*. arXiv. https://doi.org/10.48550/arXiv.1906.05799

Omefe, S., Lawal, S. A., Bello, S. F., Balogun, A. K., Taiwo, I., Ifiora, K. N. (2021). AI-Augmented Decision Support System for Sustainable Transportation and Supply Chain Management: A Review. *Communication In Physical Sciences*. 7, 4, pp. 630-642.

Lawal, S. A., Omefe, S., Balogun, A. K., Michael, C., Bello, S. F., Owen, I. T.,

Ifiora, K. N. (2021). Circular Supply Chains in the Al Era with Renewable Energy Integration and Smart Transport Networks. *Communication in Physical Sciences*, 7, 4, pp. 605-629.

Oprea, A., Li, Z., Yen, T. F., Chin, S. H., & Alrwais, S. (2015). Detection of early-stage enterprise infection by mining large-scale log data. In *2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks* (pp. 45–56). IEEE. https://doi.org/10.1109/DSN.2015.14

Qamar, S., Anwar, Z., Rahman, M. A., Al-Shaer, E., & Chu, B. T. (2017). Data-driven analytics for cyber-threat intelligence and information sharing. *Computers & Security*, *67*, pp. 35–58. https://doi.org/10.1016/j.cose.2017.02.005 *(Note: This journal did not originally list an issue number, so only volume and page range are shown.)*

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144). ACM. https://doi.org/10.1145/2939672.2939778

Scarfone, K., & Mell, P. (2007). *Guide to intrusion detection and prevention systems (IDPS)* (NIST Special Publication 800-94). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-94

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy* (pp. 108–116). SCITEPRESS. https://doi.org/10.5220/0006639801080116

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). *Intriguing properties of neural networks*. arXiv. https://doi.org/10.48550/arXiv.1312.6199

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications* (pp. 1–6). IEEE. https://doi.org/10.1109/CISDA.2009.5356528

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, 30, pp. 5998–6008). Neural Information Processing Systems Foundation.

Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, F., & Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, *1*, pp. 57–81. https://doi.org/10.1016/j.aiopen.2021.01.001 *(Note: This journal did not originally list an issue number, so only volume and page range are shown.)*

Zimmerman, C. (2014). *Ten strategies of a world-class cybersecurity operations center*. The MITRE Corporation. https://www.mitre.org/publications/technical-papers/ten-strategies-of-a-worldclass-cybersecurity-operations-center

Zliobaite˙, I., Pechenizkiy, M., & Gama, J. (2016). An overview of concept drift applications. In N. Japkowicz & J. Stefanowski (Eds.), *Big data analysis: New algorithms for a new society* (pp. 91–114). Springer. https://doi.org/10.1007/978-3-319-26989-4

**Consent for publication**

Not Applicable

**Availability of data and materials**

The publisher has the right to make the data public

**Competing interest**

Authors declared no conflict of interest.

This work was sole collaboration among all the authors

**Funding**

There is no source of external funding

**Authors Contributions**

Abdulaziz Olaleye Ibiyeye conceptualized the study, developed the analytical framework, and supervised data validation. Joy Nnenna Okolo conducted literature review, data preprocessing, and performance evaluation of AI models. Samuel Adetayo Adeniji performed model implementation, statistical analysis, visualization, and manuscript drafting. All authors reviewed, edited, and approved the final version for publication.