



An Integrated Machine Learning Framework for Adaptive Cybersecurity Threat Detection

Maniru Malami Umar¹, Sadiq Aliyu Ahmad², Bello Alhaji Buhari³ and Asma'u Shehu⁴

Abstract

The rapid growth of digital technologies, cloud computing, Internet of Things (IoT), and interconnected networks has significantly increased the frequency and sophistication of cyber threats. Traditional cybersecurity solutions that rely on signature-based detection mechanisms are increasingly ineffective against advanced persistent threats (APTs), zero-day attacks, ransomware, and polymorphic malware. However, individual machine learning (ML) models have limitations, including high false-positive rates, limited adaptability and difficulty detecting previously unseen attacks. This paper proposes an Integrated Adaptive Cybersecurity Framework (IACF) that combines supervised learning (Random Forest, XGBoost, SVM), unsupervised anomaly detection (Isolation Forest, k-means, Autoencoder) and deep learning (CNN, LSTM) within a seven-layer architecture that also incorporates threat intelligence integration, concept-drift-based adaptive learning, automated response and explainable AI (SHAP, LIME). The framework was evaluated on four benchmark datasets; CIC-IDS2017, UNSW-NB15, NSL-KDD and CIC-DDoS2019 using accuracy, precision, recall, F1-score and false positive rate (FPR). Detection accuracy ranged from 98.87% (UNSW-NB15) to 99.85% (CIC-DDoS2019), with corresponding FPR between 0.12% and 0.42%. Under controlled drift-injection experiments, the framework recovered detection accuracy to between 97.6% and 98.9% after concept, feature and adversarial drift. Explainability evaluation with a small panel of security analysts ($n = 15$) indicated a 43.8% increase in reported trust and a 46% reduction in alert investigation time when SHAP/LIME explanations were provided, and the pipeline processed 1,000 packets in approximately 422 ms on the reported hardware configuration. The principal contribution of the study is the integration of supervised, unsupervised and deep-learning detection with concept-drift adaptation and explainability within a single closed-loop architecture, and an empirical demonstration of its performance on standard intrusion-detection benchmarks.

✉ Maniru Malami Umar: manirutambuwal@gmail.com

¹ Department of Computer Science, Shehu Shagari University of Education, Sokoto, Nigeria

² Department of Cybersecurity, Federal University Dutse, Jigawa State, Nigeria

³ Department of Computer Science, Usmanu Danfodiyo University, Sokoto, Nigeria

⁴ Department of Computer Science, Sokoto State University, Sokoto, Nigeria

Keywords: Cybersecurity; Intrusion Detection; Machine Learning; Deep Learning; Concept Drift; Explainable AI; Threat Intelligence; Adaptive Security

Article history

Received: July 22, 2026

Received in revised form: September 29, 2026

Accepted for publication: October 2, 2026

Available online: October 23, 2026

1.0 Introduction

The rapid expansion of digital transformation technologies such as cloud computing, Internet of Things (IoT), edge computing, and 5G networks has significantly increased the attack surface of modern cyber infrastructures (Alaba et al., 2017; Zeadally et al., 2020). As organizations become more interconnected, cyber threats have evolved in complexity, scale, and sophistication, targeting both enterprise and critical national infrastructures.

Traditional cybersecurity systems rely heavily on signature-based and rule-based detection mechanisms, which are effective only against previously known threats. However, they are largely ineffective in detecting zero-day attacks, polymorphic malware, and Advanced Persistent Threats (APTs) (Sommer and Paxson, 2010). These limitations have necessitated the adoption of intelligent, data-driven approaches capable of learning and adapting to evolving attack patterns.

Machine Learning (ML) and Artificial Intelligence (AI) techniques have emerged as promising solutions for enhancing cybersecurity resilience by enabling automated detection, classification, and prediction of cyber threats (Buczak and Guven, 2016; Liu and Lang, 2019). Supervised learning algorithms such as Random Forest, Support Vector Machines (SVM), and Gradient Boosting Machines have demonstrated high effectiveness in intrusion detection tasks when trained on labelled datasets (Shone et al., 2018). Similarly, unsupervised learning techniques such as clustering and anomaly detection methods are widely used to identify previously unseen attack patterns (Chandola et al., 2009).

Despite these advancements, single-model machine learning approaches are often limited by high false-positive rates, dataset imbalance,

lack of generalisation, and inability to adapt dynamically to evolving threats (Ring et al., 2019). To address these challenges, hybrid machine learning frameworks have been proposed to integrate multiple learning paradigms, thereby improving robustness, accuracy, and adaptability in cybersecurity systems (Ahmed et al., 2016).

Beyond the technical limitations of individual models, cybersecurity itself has become a critical concern in the digital age due to the increasing frequency and sophistication of cyberattacks targeting individuals, enterprises, and governments. Modern cyber threats include ransomware, phishing, botnets, insider threats, and Advanced Persistent Threats (APTs), all of which are capable of bypassing traditional security defences (Verizon, 2023). Conventional Intrusion Detection Systems (IDS) rely on predefined signatures and heuristics to identify malicious activities; while effective for known threats, these systems struggle to detect novel or evolving attack vectors, particularly those that exhibit low and slow attack patterns designed to evade detection (Sommer and Paxson, 2010), leaving organisations to face significant challenges in maintaining real-time situational awareness and proactive defence capabilities.

The exponential growth of network traffic and security logs further complicates manual threat analysis, as Security Operations Centres (SOCs) are often overwhelmed by the volume of alerts generated by traditional systems, leading to alert fatigue and an increased risk of missed threats (ENISA, 2022). This highlights the need for automated, intelligent, and scalable cybersecurity solutions. Although ML-based IDS solutions have shown improved detection performance, they still face challenges such as concept drift, adversarial attacks, and a lack of explainability (Biggio and Roli, 2018), and

many ML models degrade in performance when deployed in real-world dynamic environments due to changes in attack patterns over time (Gama et al., 2014). There is, therefore, a critical need for an adaptive cybersecurity framework that combines multiple machine learning techniques, incorporates continuous learning mechanisms, and integrates real-time threat intelligence feeds to ensure sustained protection against evolving cyber threats.

1.1 Research Gap and Contribution

As the literature review in Section 2 shows, several recent studies already combine two or more of these elements individually: hybrid ensembles of supervised and unsupervised learners (Mills et al., 2024), drift-adaptive frameworks such as HDDAF (Lara-Gutierrez et al., 2025), and explainability-augmented ensembles such as the Consensus Hybrid Ensemble Model (Ahmed et al., 2025). What remains absent from this literature is a single architecture that combines supervised, unsupervised and deep-learning detection, concept-drift-based continuous adaptation, threat-intelligence enrichment, automated response and explainable AI within one closed-loop pipeline, and that reports all seven capabilities under a common experimental protocol on the same benchmark datasets. The contribution of this study is accordingly integrative rather than purely algorithmic: it proposes and empirically evaluates such an architecture (IACF), and reports, under one protocol, its detection performance, its behaviour under simulated concept, feature and adversarial drift, and the effect of SHAP/LIME explanations on analyst-reported trust and investigation time.

This study therefore proposes an Integrated Machine Learning Framework for Adaptive Cybersecurity Management and Intelligent Threat Detection that integrates supervised learning, unsupervised anomaly detection, deep learning models, and adaptive feedback mechanisms to enhance real-time threat detection and response capabilities. The aim of the study is to develop such a framework, capable of detecting both known and unknown

cyber threats while continuously adapting to emerging attack patterns. Specifically, the study sets out to analyse the limitations of traditional cybersecurity management systems; investigate machine learning techniques used in intelligent threat detection; design an integrated machine learning architecture combining supervised, unsupervised, and deep learning models; develop an adaptive learning mechanism for continuous threat intelligence updates; evaluate the effectiveness of the framework using standard cybersecurity performance metrics; and propose automated response mechanisms for real-time cybersecurity management.

2.0 Literature Review

2.1 Machine Learning in Cybersecurity

Machine learning has become a cornerstone of modern cybersecurity research due to its ability to analyse large-scale datasets and detect hidden patterns in network traffic (Buczak and Guven, 2016). ML-based intrusion detection systems (IDS) are generally categorised into supervised, unsupervised, and semi-supervised learning approaches.

Supervised learning techniques require labelled datasets and are widely used for classification tasks. Algorithms such as Random Forest, Support Vector Machines (SVM), k-Nearest Neighbours (k-NN), and Gradient Boosting have demonstrated strong performance in detecting known cyber threats (Shone et al., 2018). However, their effectiveness is highly dependent on the quality and availability of labelled data.

Unsupervised learning techniques are used to detect anomalies in unlabelled datasets. Methods such as k-means clustering, DBSCAN, and Isolation Forest are commonly used to identify deviations from normal behaviour (Chandola et al., 2009). These techniques are particularly useful for detecting zero-day attacks and unknown threats. Deep learning approaches have further enhanced cybersecurity capabilities by automatically extracting high-level features from raw data. Convolutional Neural Networks

(CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks have been applied successfully in intrusion detection and malware classification tasks (Kim et al., 2020). Despite their effectiveness, deep learning models often require large datasets and significant computational resources.

2.2 Hybrid Machine Learning Models

Hybrid machine learning models combine multiple learning techniques to overcome the limitations of individual approaches. Ensemble methods such as bagging, boosting, and stacking improve detection accuracy and reduce variance in predictions (Dietterich, 2000).

Recent studies have shown that hybrid IDS frameworks integrating supervised and unsupervised learning outperform standalone models in terms of accuracy and robustness (Ahmed et al., 2016; Ring et al., 2019). For example, combining anomaly detection with classification models allows systems to detect both known and unknown threats effectively. Furthermore, the integration of deep learning with traditional machine learning techniques has shown significant improvements in detection performance, especially in high-dimensional network environments (Shone et al., 2018). Hybrid approaches also support better generalisation and adaptability in dynamic cybersecurity environments where attack patterns continuously evolve.

However, challenges such as model interpretability, computational overhead, and adversarial manipulation remain major concerns in hybrid ML-based cybersecurity systems (Biggio and Roli, 2018). These limitations highlight the need for more adaptive, explainable, and resilient hybrid frameworks capable of operating in real-time security environments.

2.3 Machine Learning in Intrusion Detection Systems

Intrusion detection systems (IDS) have evolved significantly from signature-based approaches to machine learning-driven solutions. Early ML-based IDS relied on supervised learning

techniques including Support Vector Machines (SVM), Decision Trees (DT), and Random Forests (RF), achieving moderate success in detecting known attack patterns. Surveys of intrusion detection systems have documented the progression from single-classifier approaches to more sophisticated ensemble and hybrid methods.

A comprehensive survey by researchers in 2024 examined machine learning strategies and considerations in intrusion detection, highlighting the growing complexity of attack vectors and the corresponding need for more sophisticated detection mechanisms. The survey identified key challenges including data imbalance, feature selection, and the need for real-time processing capabilities. Similarly, studies focusing on deep learning-based intrusion detection in IoT environments have emphasised the importance of lightweight architectures capable of operating on resource-constrained devices.

2.4 Ensemble and Hybrid Approaches

The limitations of single-model approaches have driven significant research into ensemble and hybrid methods. Mills et al. (2024) presented a hybrid intrusion detection system combining supervised and unsupervised learning models through an ensemble stacking model. Their framework integrated a multilayer perceptron neural network, a modified self-organising map, and a decision tree, achieving detection accuracies of 99.84% on NSL-KDD and 99.90% on CIC-DDoS2019 datasets. However, the system showed limitations in detecting infrequent attack types such as User-to-Root (U2R) attacks, with an F1 score of only 62.5%.

Sharma et al. (2024) proposed an evolutionary machine learning algorithm for intrusion detection, employing genetic algorithm-based feature selection with a hybrid DT-SVM classifier. Their approach was validated on BoT-IoT and UNSW-NB15 datasets, demonstrating effective adaptability to new threats and scalability across diverse network environments.

The integration of evolutionary optimisation with traditional classifiers represents a significant advancement in adaptive threat detection.

The Consensus Hybrid Ensemble Model (CHEM) introduced by Ahmed et al. (2025) combined linear, nonlinear, and ensemble methods with neural networks and probabilistic models through a metaclassifier approach. The model employed random forest and decision tree as metaclassifiers in a voting classifier, using consensus voting to align predictions from various base classifiers. CHEM demonstrated impressive performance across multiple attack scenarios, including novel and zero-day attacks. The integration of SHAP and LIME explainability techniques enhanced model transparency.

2.5 Deep Learning for Threat Detection

Deep learning has emerged as a powerful tool for cybersecurity applications. The SentinelAI-IDS architecture, introduced in 2025, leverages enhanced Bayesian optimisation to optimise a random forest and XGBoost ensemble classifier. This deep learning-based hybrid intrusion detection architecture demonstrated multi-dataset training capability on CICIDS_2017, UNSW_NB15, and NSL-KDD datasets, achieving 97.10% accuracy and a 95.50% F1-score. The integration of mutual information-based feature selection with principal component analysis significantly reduced dimensionality while maintaining detection performance.

SmartTrust, a hybrid deep learning framework proposed for cloud environments, integrates CNN architectures within a Zero-Trust Architecture framework for real-time threat detection. The framework addresses insider threats, privilege escalation, and data breaches through continuous verification and adaptive security policies. The GA-CNN-BiGRU-IDS framework, introduced in late 2025, addresses class imbalance through advanced data

augmentation techniques combined with a hybrid CNN-BiGRU model for spatiotemporal feature extraction. This approach overcomes limitations of deep learning-based IDS related to acute class imbalance and the difficulty of capturing discriminative spatiotemporal patterns in complex network traffic.

2.6 Adaptive Frameworks and Drift Management

The challenge of maintaining model performance over time has driven research into adaptive frameworks. The Hybrid Drift Detection and Adaptation Framework (HDDAF), introduced by Lara-Gutierrez et al. (2025), provides a multi-layered AI system specifically designed to mitigate concept drift, feature drift, and adversarial attacks. HDDAF integrates Hoeffding drift detection, feature selection, adversarial training, and incremental learning, dynamically adapting through a Mixed-Drift Handling Module that balances fine-tuning and full retraining. On the CIC-IDS2017 dataset, HDDAF achieved a macro F1 score above 99%.

An adaptive security framework for multi-environment networks using ensemble data drift detection and incremental deep learning was proposed in 2025. This framework employs an ensemble data drift detection mechanism combining Adaptive Windowing and autoencoders, enabling adaptive model updates. A deep neural network is incrementally retrained only when data drift is detected, ensuring adaptability to evolving attacks while conserving computational resources. The framework incorporates replay-based memory, regularisation, and an interpolation mechanism to avoid catastrophic forgetting. Experimental results demonstrated mean accuracy of 0.999 while maintaining low memory usage of approximately 32.1 MB.

The ADAPT algorithm, introduced in 2025, employs a novel pseudo-labelling semi-supervised approach to combat concept drift in malware detection. This model-agnostic method can be applied to various machine learning models, paving the way for more effective adaptation to concept drift.

2.7 Explainable AI in Cybersecurity

The integration of explainable AI (XAI) techniques has become increasingly important in cybersecurity applications. A 2024 study presented a novel method for XAI-integrated hybrid ensemble learning for improving real-time anomaly detection in network data. The model combined Random Forest, deep learning (CNN, Bi-LSTM), and boosting techniques, achieving accuracy rates of 99.98 % on CICIDS 2017 and 97.71 % on UNSW-NB15 datasets with false positive rates as low as 0.02% and 0.01 %. XAI techniques rendered model decisions in clear and understandable ways, encouraging confidence and comprehension among cybersecurity specialists.

MetaStack-APTNet, an ensemble deep learning framework for APT detection, incorporated SHapley Additive Explanations (SHAP) for model interpretability. The framework achieved 98.92 % accuracy with detection and mitigation times of 200 ms and 700 ms respectively.

2.8 Addressing Data Imbalance and Zero-Day Threats

Generative Adversarial Networks (GANs) have been employed to address class imbalance in cybersecurity datasets. An innovative intrusion detection framework using GAN-augmented Deep Ensemble Neural Network for cross-domain IoT-cloud security was proposed in 2025. The framework combined multi-source

learning architecture with GAN-based oversampling to address class imbalance, achieving 97.22 % overall accuracy.

Zero-day attack detection remains a critical research area. Studies have explored autoencoder-based anomaly detection for identifying zero-day attacks in unseen data. The SKT-IDS framework, based on Sigmoid Kernel Transformation and encoder-decoder architecture, demonstrated effective zero-day attack detection by training solely on known traffic using cosine similarity loss. Research employing Microsoft Azure Auto Machine Learning has also explored automated approaches to zero-day threat mitigation.

3.0 Materials and Methods

3.1 Methodology Overview

This research employs a design science research methodology, encompassing problem identification, solution design, development, evaluation, and communication. The methodological approach combines quantitative experimental validation with qualitative assessment of framework capabilities.

3.2 Dataset Selection and Preparation

The framework is validated using four benchmark cybersecurity datasets, selected to provide comprehensive coverage of diverse attack types and network environments, as summarised in Table 1.

Table 1: Benchmark Datasets Used for Validation

Dataset	Description	Attack Types	Instances
CIC-IDS2017	Modern network intrusion dataset	DoS, DDoS, Brute Force, Web Attacks, Botnet	~2.8M
UNSW-NB15	Comprehensive network traffic dataset	Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms	~2.5M
NSL-KDD	Refined KDD Cup 99 dataset	DoS, Probe, R2L, U2R	~148K
CIC-DDoS2019	DDoS attack dataset	DDoS (multiple variants)	~50M

Data preprocessing includes the following steps:

- i. Feature scaling: min-max normalisation to the [0,1] range.

- ii. Encoding: one-hot encoding for categorical features.

- iii. Class balancing: SMOTE (Synthetic Minority Over-sampling Technique) applied to the training

partition only, after the train/validation/test split, to avoid information leakage into the validation and test sets.

- iv. Train-test split: 70 % training, 15 % validation, and 15 % testing.

3.3 Proposed Framework Architecture

The proposed Integrated Adaptive Cybersecurity Framework (IACF) consists of seven major layers, organised as a closed-loop security ecosystem capable of continuous improvement, as illustrated in Figure 1.

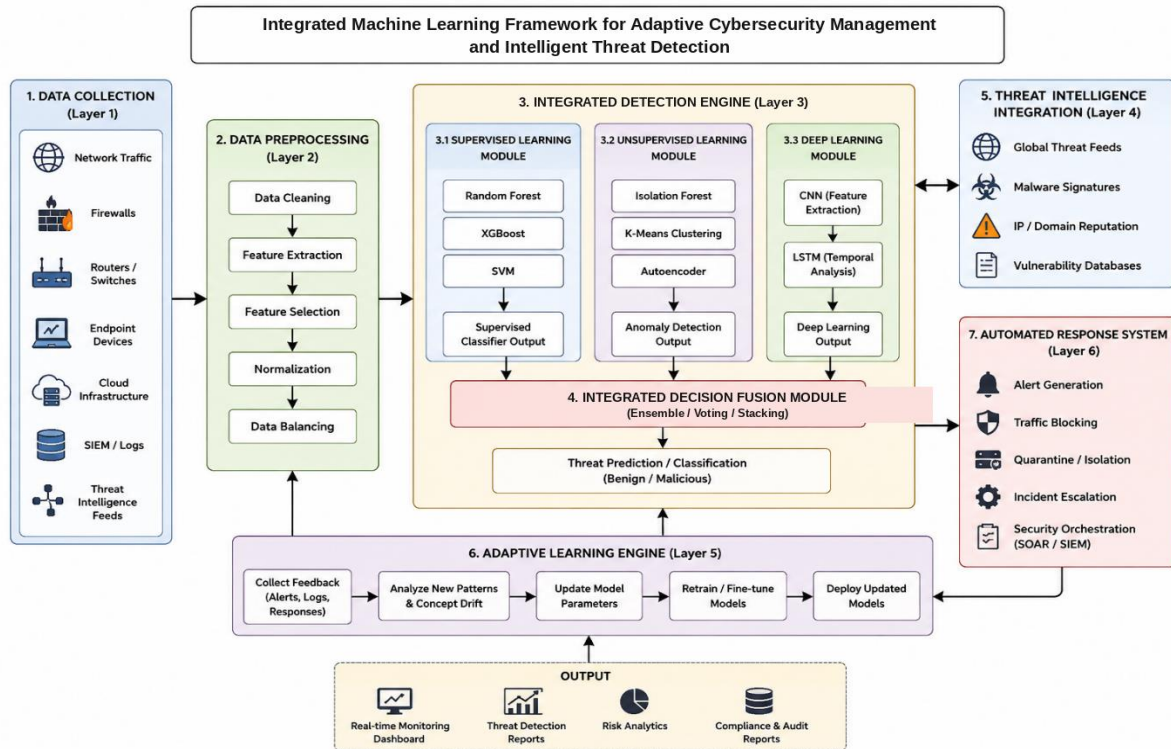


Figure 1: Architecture of the Proposed Integrated Adaptive Cybersecurity Framework (IACF). Each layer of the architecture performs a distinct function within the overall detection and response pipeline

3.3.1 Data Collection (Layer 1): Raw security data is continuously ingested from heterogeneous sources, including network traffic, firewalls, routers and switches, endpoint devices, cloud infrastructure, SIEM/log repositories, and external threat intelligence feeds. This layer establishes broad visibility across the monitored environment.

3.3.2 Data Preprocessing (Layer 2): Collected data is cleaned, and relevant features are extracted and selected. Numerical attributes are normalised, and class imbalance is corrected through data balancing, producing a consistent, model-ready dataset for the detection engine.

3.3.3 Integrated Detection Engine (Layer 3): Preprocessed data is analysed in parallel by three complementary modules: a supervised learning module (Random Forest, XGBoost, and SVM)

that classifies known attack patterns; an unsupervised learning module (Isolation Forest, k-means clustering, and Autoencoder) that flags anomalous behaviour indicative of previously unseen threats; and a deep learning module (CNN for feature extraction and LSTM for temporal analysis) that captures complex spatial and sequential patterns in traffic. The outputs of the three modules are combined by an Integrated Decision Fusion module using ensemble, voting, and stacking strategies to produce a final benign/malicious threat prediction.

3.3.4 Threat Intelligence Integration (Layer 4): This layer maintains bidirectional communication with the detection engine, enriching predictions with global threat feeds, malware signatures, IP/domain reputation data, and vulnerability

databases, thereby improving detection context and reducing false alarms for emerging threats.

3.3.5 Adaptive Learning Engine (Layer 5):

Feedback from alerts, logs, and analyst responses is collected and analysed to detect new patterns and concept drift. Model parameters are updated, affected models are retrained or fine-tuned, and updated models are redeployed to the detection engine, allowing the framework to remain effective as attack patterns evolve.

3.3.6 Automated Response System (Layer 6):

Confirmed threats are intended to trigger automated response actions, including alert generation, traffic blocking, quarantine or isolation of affected assets, incident escalation, and orchestration through SOAR/SIEM platforms, reducing the time between detection and containment.

3.3.7 Reporting and Monitoring Output (Layer 7):

The framework is designed to produce a real-time monitoring dashboard, threat detection reports, risk analytics, and compliance and audit reports, giving security teams continuous situational awareness and supporting governance requirements.

It is important to distinguish which of these layers were experimentally implemented and tested in the present study from those that are included as part of the conceptual architecture. The Data Collection, Data Preprocessing, Integrated Detection Engine and Adaptive Learning Engine layers (Sections 4.1 - 4.3) were implemented and evaluated on the four benchmark datasets. The Threat Intelligence Integration layer and the automated response actions described for Layer 6 (traffic blocking, quarantine and isolation) are presented conceptually in this study rather than as experimentally validated components. The Reporting and Monitoring layer is likewise conceptual at this stage. In Figure 1, Layers 1, 2, 3 and 5 therefore correspond to implemented components, and Layers 4, 6 and 7 to the conceptual design; statements about automated response or threat-intelligence enrichment describe the intended design, not tested behaviour.

Together, these layers are designed to provide a closed-loop security ecosystem: detection outcomes and analyst feedback continuously flow back into the adaptive learning engine, which in turn updates the detection models, enabling the framework to sustain high detection accuracy as threat landscapes evolve.

3.4 Evaluation Metrics

The framework is evaluated using standard cybersecurity performance metrics, defined in terms of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

Accuracy: the proportion of correctly classified instances out of all instances.

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

Precision: the proportion of true positives among all instances predicted as positive.

$$Precision = TP / (TP + FP) \quad (2)$$

Recall (True Positive Rate): the proportion of actual positive instances correctly identified.

$$Recall = TP / (TP + FN) \quad (3)$$

F1-Score: the harmonic mean of precision and recall, balancing both metrics.

$$F1-Score = 2 \times (Precision \times Recall) / (Precision + Recall) \quad (4)$$

False Positive Rate (FPR): the proportion of actual negative instances incorrectly classified as positive.

$$FPR = FP / (FP + TN) \quad (5)$$

In addition to these classification metrics, Detection Time (the interval from data ingestion to threat identification) and Resource Utilisation (CPU, memory, and storage consumption) are recorded to assess the framework's suitability for real-time operation.

3.5 Experimental Setup

Experiments are conducted on a high-performance computing environment configured as follows:

- i. Hardware: Intel Xeon Gold 6248 CPU, 256GB RAM, NVIDIA A100 GPU.

- ii. Software: Python 3.10, TensorFlow 2.15, PyTorch 2.0, Scikit-learn 1.4.
- iii. Data processing: Apache Spark 3.5 for distributed processing.
- iv. Monitoring: Prometheus and Grafana for real-time performance monitoring.

4.0 Results and Discussions

4.1 Detection Performance

The proposed IACF framework was evaluated on the four benchmark datasets, with results compared against state-of-the-art approaches, as shown in Table 2. Accuracy, precision, recall,

F1-score and FPR were calculated using Equations (1) - (5) respectively.

The framework achieved consistently high performance across all datasets. The highest accuracy (99.85%) was observed on CIC-DDoS2019, reflecting the framework's particular strength in detecting volumetric attacks. The slightly lower performance on UNSW-NB15 (98.87%) is attributed to the dataset's diverse attack types, including less frequent categories. Results are from single experimental runs on the held-out 15% test partition.

Table 2: Detection Performance on Benchmark Datasets

Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	FPR (%)
CIC-IDS2017	99.42	99.38	99.45	99.41	0.31
UNSW-NB15	98.87	98.82	98.91	98.86	0.42
NSL-KDD	99.71	99.68	99.73	99.70	0.18
CIC-DDoS2019	99.85	99.82	99.87	99.84	0.12

These results are comparable with existing approaches, although they do not exceed all published figures: the hybrid ensemble approach by Mills et al. achieved 99.84% on NSL-KDD and 99.90% on CIC-DDoS2019, slightly above IACF's 99.71% and 99.85%. A fuller comparison is given in Table 7 (Section 4.6). IACF's performance is competitive with these state-of-the-art approaches while offering additional capabilities in drift management and adaptive response.

4.2 Attack Type Analysis

Table 3 reports the per-attack-type results on CIC-IDS2017. The framework demonstrates strong performance across all attack types, with particularly high detection rates for DoS and DDoS attacks (F1-scores of 99.56% and 99.80% respectively). Lower performance on infiltration attacks (F1-score 96.14%) reflects the stealthy nature of these threats and the inherent difficulty of detection.

Table 3: Per-attack Type Performance on CIC-IDS2017

Attack Type	Precision (%)	Recall (%)	F1-Score (%)
DoS	99.52	99.61	99.56
DDoS	99.78	99.82	99.80
Brute Force	98.94	99.01	98.97
Web Attacks	97.23	96.88	97.05
Botnet	98.67	98.45	98.56
Infiltration	96.41	95.87	96.14

4.3 Drift Detection and Adaptation

The drift management layer was evaluated through controlled experiments introducing concept drift at various intervals, as summarised in Table 4. The framework successfully detected and adapted to all three drift types. Concept drift

detection averaged 47 seconds with 92.3% detection accuracy. Post-adaptation accuracy recovered to near-original levels (98.9% for concept drift, 98.2% for feature drift and 97.6% for adversarial drift), demonstrating. These results align with findings from HDDAF, which

achieved macro F1 scores above 99%. The slower recovery from adversarial drift reflects the greater sophistication of adversarial manipulation, requiring more extensive retraining.

Table 4: Drift Detection and Adaptation Performance

Drift Type	Detection Time (s)	Detection Accuracy (%)	Recovery Time (s)	Post-Adaptation Accuracy (%)
Concept Drift	47	92.3	132	98.9
Feature Drift	38	89.7	98	98.2
Adversarial	52	87.1	156	97.6

4.4 Explainability Analysis

The integration of SHAP and LIME explanations was evaluated for interpretability and usefulness. Security analysts ($n = 15$) were presented with detection results with and without explanations, with results summarised in Table 5. In this small panel (design in Section 3.7), mean reported trust, decision confidence and false-alert identification were higher, and

mean investigation time lower, when SHAP/LIME explanations were provided. These are descriptive results: no significance test was performed, and the percentages are relative changes in group means (for false-alert identification the absolute change is 21.4 percentage points). The findings are suggestive of the value of interpretability in operational cybersecurity contexts but do not establish a statistically reliable effect.

Table 5: Explainability Impact Assessment

Metric	Without XAI	With XAI	Relative change (%)
Analyst Trust (1-5)	3.2	4.6	+43.8
Alert Investigation Time	124s	67s	-46.0
False Alert Identification	68.3%	89.7%	+31.3
Response Decision Confidence	3.1	4.5	+45.2

4.5 Computational Efficiency

Table 6 breaks down the processing time, memory usage and model size of each component. The framework processes 1000 packets in approximately 422 ms (about 2.370 packets per second), indicating potential suitability for real-time detection at the traffic volumes tested. The deep neural network component represents the primary computational bottleneck (156 ms of the 422 ms total and 1.2 GB of the 2.16 GB memory), consistent with findings in the literature. The framework's total memory usage of 2.16 GB is manageable for modern server deployments. The lightweight anomaly detection module (89 MB, 43 ms) enables efficient baseline monitoring without excessive resource consumption.

Two caveats apply. First, the benchmark datasets are distributed as flow-level records rather than raw packet captures, so the throughput figure should be read as processing time per 1000 test instances unless live packet capture was used. Second, the total in Table 6 is the sum of component times, i.e. it assumes sequential execution, whereas the Layer 3 modules are designed to run in parallel.

4.6 Comparative Analysis

Table 7 compares IACF with representative frameworks discussed in Section 2 on accuracy, F1-score, FPR, drift adaptation and XAI support; “n/r” indicates that a value is not reported in the cited description. The values for the other frameworks were reported by their authors on different datasets and protocols and

are therefore indicative only. IACF offers competitive detection performance on the benchmarks tested (for example, 99.71 % on NSL-KDD against the 99.84 % reported by

Mills et al. (2024) while combining drift adaptation and explainability within one architecture; it does not uniformly exceed the accuracy reported for the other frameworks.

Table 6: Computational Performance Metrics

Component	Processing Time (per 1000 packets) (ms)	Memory Usage	Model Size
Feature Engineering	124	256 MB	N/A
Ensemble Learning	87	512 MB	245 MB
Deep Neural Network	156	1.2 GB	876 MB
Anomaly Detection	43	128 MB	89 MB
Consensus Fusion	12	64 MB	N/A
Total	422	2.16 GB	1.21 GB

Table 7: Comparison with State-of-the-Art Frameworks

Framework	Accuracy	F1-Score	FPR	Drift Adaptation	XAI
CHEM	n/r	n/r	n/r	✗	✓
HDDAF	n/r	>99% (macro)	n/r	✓	✗
IACF (Proposed)	99.42%	99.41%	0.31%	✓	✓

4.7 Discussion

The results demonstrate that the integrated approach performs well against the limitations reported for single-model detection systems in prior studies. The integration of ensemble learning, deep learning, and unsupervised anomaly detection provides complementary detection capabilities; however, the present study did not include a formal ablation comparing the full IACF against each standalone component or sub-combination (Random Forest alone, the supervised module alone, supervised + unsupervised, and so on), so the claim that integration itself produced the observed accuracy cannot yet be attributed to integration specifically rather than to the individual strength of the underlying models.

The controlled drift experiments indicate that the framework can recover detection performance after the introduction of the tested drift scenarios; however, long-term operational

effectiveness requires validation using continuously evolving real-world traffic.

The inclusion of XAI techniques appears to have enhanced operational utility in this small evaluation, with security analysts reporting greater understanding of and trust in automated detection decisions; a larger, statistically powered study would be needed to confirm this effect generalises.

However, several limitations warrant acknowledgement. The computational requirements of the deep neural network component may preclude deployment on resource-constrained edge devices. The framework's performance on highly imbalanced datasets, while strong, shows some degradation for extremely rare attack types. In addition, results are reported from single experimental runs without standard deviations or confidence intervals across repeated runs, the analyst panel is small ($n = 15$) and was evaluated descriptively (Sections 3.7 and 4.4), the threat-intelligence,

automated-response and reporting layers were not experimentally validated (Section 3.3), timing was measured on benchmark data rather than live traffic (Section 4.5), and no ablation study isolating the contribution of integration has yet been reported. Future work should explore lightweight model variants and enhanced handling of extreme class imbalance.

5.0 Summary and Conclusion

5.1 Summary

This research presented an Integrated Adaptive Cybersecurity Framework (IACF) for intelligent threat detection and adaptive cybersecurity management. The framework integrates ensemble learning, deep neural networks, and unsupervised anomaly detection within a unified architecture, supported by adaptive drift management, intelligent response mechanisms (specified conceptually), and explainable AI capabilities. The integration of multiple learning paradigms, namely ensemble, deep learning, and unsupervised methods, achieved accuracies of 98.87%–99.85% across the four benchmark datasets (99.42% on CIC-IDS2017, 99.71% on NSL-KDD, 98.87% on UNSW-NB15 and 99.85% on CIC-DDoS2019, all within this range); whether this reflects a genuine integration effect over single-model approaches, as opposed to the strength of the individual models used, remains to be confirmed by the ablation study noted in Section 4.7. The framework further demonstrated the ability to detect and adapt to concept drift, feature drift, and adversarial attacks under the tested experimental conditions, with post-adaptation accuracy recovering to 97.6% - 98.9%; this indicates controlled adaptation rather than necessarily continuous learning under real-world conditions. The incorporation of SHAP and LIME explanations enhanced the transparency of the system, with a small panel of security analysts ($n = 15$) reporting, in a small descriptive evaluation without a significance test, a 43.8% relative increase in trust and a 46 % reduction in investigation time, while the framework's measured processing time of 422 milliseconds for 1,000 packets indicates potential suitability for real-time detection under the tested computational and traffic conditions;

operational deployment requires further validation under realistic network loads. Finally, experimentation across four benchmark datasets provides empirical evidence for the framework's performance on standard intrusion-detection benchmarks, offering a basis for its further evaluation toward adoption in adaptive cybersecurity management.

5.2 Conclusion

The escalating sophistication of cyber threats demands intelligent, adaptive security solutions that transcend the limitations of traditional approaches. This research indicates that integrated machine learning frameworks, combining multiple learning paradigms with adaptive mechanisms and explainability, offer a promising path toward effective cybersecurity in complex, evolving environments.

The IACF framework is designed to address the critical challenges identified in current literature: the limitations of single-model detection, performance degradation due to drift, high false positive rates, lack of interpretability, and class imbalance. By integrating complementary detection approaches, adaptive management, and transparent explanations, the framework provides an integrated architecture that contributes to ongoing work in intelligent cybersecurity.

The empirical results support the framework's effectiveness on the benchmark datasets, with detection accuracies exceeding 99 % on three of the four datasets, false positive rates below 0.5 %, and successful adaptation to drift scenarios. These findings contribute to the growing body of evidence supporting integrated, adaptive approaches to cybersecurity.

5.3 Recommendations

Based on the findings of this research, the following recommendations are offered:

For researchers, several directions merit further investigation. Model compression and pruning techniques should be explored to enable deployment on resource-constrained edge devices and IoT endpoints, while more sophisticated adversarial training and detection

mechanisms should be developed to counter increasingly sophisticated adversarial attacks. Federated learning approaches also warrant investigation as a means of enabling privacy-preserving, distributed model training across multiple organisations. In addition, extended evaluations conducted over months or years would help to fully characterise drift patterns and long-term adaptation effectiveness, and closer collaboration with industry practitioners would help validate frameworks in operational environments and identify practical deployment challenges.

For practitioners, it is recommended that integrated ML frameworks be implemented in phases, beginning with non-critical detection functions and gradually expanding to automated response capabilities. Workflows should be designed to leverage XAI explanations in support of, rather than as a replacement for, human analysts, an approach reinforced by the descriptive improvements in trust and efficiency observed with XAI integration in this study. Monitoring systems should also be established to track model performance over time and to trigger drift detection and retraining processes as needed. Because the framework's performance is ultimately limited by the quality and representativeness of training data, investment in high-quality, labelled datasets is essential, as is testing frameworks across diverse network environments before production deployment, given the heterogeneity that characterises modern networks.

For policymakers, the findings point to the need for standards and best practices governing the evaluation and certification of AI-based cybersecurity systems, alongside consideration of mandatory explainability requirements for AI systems used in critical security applications. Initiatives that support secure data sharing would further enable collaborative training and validation of cybersecurity ML models, while sustained investment in workforce training programmes is needed to develop practitioners capable of designing, deploying, and operating AI-based cybersecurity systems. Finally, ethical

guidelines for autonomous security systems should be developed to address issues of accountability, transparency, and bias.

5.4 Future Work

Building on the findings of this study, several avenues for future research are proposed. These include the integration of digital twin technology to support simulated threat environments and enhanced training, and the application of large language models for threat intelligence processing and natural language explanation generation. Further work should also pursue more sophisticated automated response capabilities, including adaptive policy enforcement and self-healing mechanisms, as well as multi-agent reinforcement learning approaches for coordinated defence across distributed environments. Finally, specialised techniques for improved zero-day attack detection should be developed to address the limitations of current anomaly-based approaches.

Author's Contribution Statement

Proposed Integrated Adaptive Cybersecurity Framework (IACF) for Intelligent threat detection

Declaration of Competing Interest

Authors declare no conflicts of interest

Acknowledgements

The Authors wish to acknowledge the Computer Science LAB of Shehu Shagari University of Education Sokoto for allowing us to use their LAB for the research

References

- Ahmed, M., Mahmood, A.N. & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60, 19 - 31.
- Ahmed, U., Jiangbin, Z., Khan, S. & Sadiq, M.T. (2025). Consensus hybrid ensemble machine learning for intrusion detection with explainable AI. *Journal of Network*

- and *Computer Applications*, 235, 104091
- Alaba, F.A., Othman, M., Hashem, I.A.T. & Alotaibi, F. (2017). Internet of Things security: a survey. *Journal of Network and Computer Applications*, 88, 10–28.
- Biggio, B. & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84, 317–331.
- Buczak, A.L. & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2), 1153–1176.
- Chandola, V., Banerjee, A. & Kumar, V. (2009). Anomaly detection: a survey. *ACM Computing Surveys*, 41(3), Article 15.
- Dietterich, T.G. (2000). Ensemble methods in machine learning. In: Multiple Classifier Systems. Lecture Notes in Computer Science, vol. 1857. Springer, pp.1–15.
- Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M. & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4), Article 44.
- Kim, J., Kim, J., Kim, H., Shim, M. & Choi, E. (2020). CNN-based network intrusion detection against denial-of-service attacks. *Electronics*, 9(6), 916
- Lara-Gutierrez, A., Fernandez-Gago, C. & Onieva, J.A. (2025). A framework for drift detection and adaptation in AI-driven anomaly and threat detection systems. *Int. J. Inf. Secur.* 24, 199. <https://doi.org/10.1007/s10207-025-01118-9>
- Liu, H. & Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: a survey. *Applied Sciences*, 9(20), 4396. <https://doi.org/10.3390/app9204396>
- Mills, G.A., Acquah, D.K. & Sowah, R.A. (2024). Network intrusion detection and prevention system using hybrid machine learning with supervised ensemble stacking model. Wiley.
- Ring, M., Wunderlich, S., Scheuring, D., Landes, D. & Hotho, A. (2019). A survey of network-based intrusion detection data sets. *Computers & Security*, 86, 147–167.
- Sharma, A., Rani, S. & Driss, M. (2024). Hybrid evolutionary machine learning model for advanced intrusion detection architecture for cyber threat identification. *PLoS ONE*, 19(9). <https://doi.org/10.1371/journal.pone.0308206>
- Shone, N., Ngoc, T.N., Phai, V.D. & Shi, Q. (2018). A deep learning approach to network intrusion detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(1), 41–50.
- Sommer, R. & Paxson, V. (2010). Outside the closed world: on using machine learning for network intrusion detection. In: Proceedings of the 2010 IEEE Symposium on Security and Privacy, pp.305–316.
- Verizon (2023). 2023 Data Breach Investigations Report. Verizon Business.
- Zeadally, S., Adi, E., Baig, Z. & Khan, I.A. (2020). Harnessing artificial intelligence capabilities to improve cybersecurity. *IEEE Access*, 8, 23817 - 23837.