© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

Comparison of Selected Machine Learning Techniques in Cyberattack Anomaly Detection

DORCAS ATINUKE ADEDOKUN[1], WASIU OLADIMEJI ISMAILA[2], SIMEON AYOADE ADEDOKUN[3], ELIZABETH A. AMUSAN[4], FOLASADE MUIBAT ISMAILA[5]

[1,2,3] *Department of Computer Science, Ladoke Akintola University of Technology, Ogbomoso, Nigeria.*

[4] *Department of Cybersecurity, Ladoke Akintola University of Technology, Ogbomoso, Nigeria.*

[5] *Department of Information Technology, University of Ilesa, Osun State, Nigeria*.

Abstract- ***The digital age has ushered in unprecedented connectivity and technological advancement, which have also introduced a surge in sophisticated and frequent cyber threats. To safeguard systems, anomaly detection has become a cornerstone of cybersecurity, enabling the identification of deviations from normal system behaviour. This study presents a comparative analysis of three machine learning techniques—Isolation Forest, Long Short-Term Memory (LSTM), and Q-Learning—for cyberattack anomaly detection. The study designed and implemented a system using the CICIDS-2017 dataset (2,830,743 records) in Python, preceded by data preprocessing and feature engineering. Evaluation metrics, including Accuracy, F1-Score, and error rates (FPR, FNR) revealed a clear performance hierarchy. The LSTM model proved superior, achieving a near-perfect Accuracy of 99.53% with minimal errors (FPR: 0.35%, FNR: 0.50%). Q-Learning showed strong, adaptive potential, recording an Accuracy of 92.80% and an F1-Score of 90.25%, though with higher error rates (FPR: 8.58%). Conversely, the unsupervised Isolation Forest was inadequate for this labeled task, with metrics around 50%. The findings establish LSTM as ideal for maximum accuracy, Q-Learning as a viable option for dynamic environments, and highlight the limitations of simple unsupervised methods on complex security datasets.***

Index Terms- ***Cyberattack, Anomaly, Detection, Machine, Learning, Isolation Forest, Q-Learning, LSTM, Long Short-Term, Memory.***

I. INTRODUCTION

The digital age, marked by exponential growth in interconnected systems, has led to a surge in sophisticated cyberattacks, posing significant financial and operational risks (Clarke et al., 2018; Kshetri, 2017). Traditional security methods, such as signature-based detection, are struggling to keep pace, frequently failing against zero-day and novel attacks while generating high false positive rates (Chandola et al., 2009; Modi et al., 2017). Machine learning (ML) offers a promising alternative for anomaly detection by adapting to evolving threats and handling complex data (Al-Shaymaa et al., 2019). This research focuses on comparing three diverse ML paradigms—Isolation Forest (IF), Long Short-Term Memory (LSTM), and Q-Learning—to identify the optimal approach for enhancing modern cybersecurity defenses.

The core research problem stems from the inability of traditional security systems to scale, integrate data, and respond effectively to the speed and volume of modern cyber threats, leading to severe vulnerabilities. While Machine Learning (ML) offers scalable solutions, the diverse performance of various algorithms creates a guidance gap for security professionals. The study is justified by the critical need for real-time analytics and advanced anomaly detection capabilities to combat the escalating costs and risks of cybercrime and establish a more proactive, holistic, and optimal cybersecurity posture.

This study carried out a comparative analysis of Isolation Forest, Long Short-Term Memory, and Q-learning machine-learning techniques across key metrics, including Accuracy, Precision, Recall, F1-score, False Positive Rate (FPR), and False Negative Rate (FNR). To achieve this, the specific objectives are to: design a cyberattack anomaly detection system using the three selected techniques; implement the designed system using the Python programming language; evaluate and compare their performances using metrics. The study's scope involved analyzing these three techniques in high-volume data environments, covering the full pipeline from data collection and preprocessing (cleaning, feature engineering) to training the algorithms and assessing their performance for timely threat detection.

© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

II. RELATED WORKS

The development of robust anomaly detection systems for cybersecurity is critical as cyber threats become increasingly sophisticated. This literature review examines significant studies on anomaly detection in cybersecurity, focusing on methodologies, algorithms, and their limitations. Key metrics, research gaps, and suggestions are highlighted, providing a foundation for advancing streaming machine learning-based cyberattack anomaly detection systems.

Early intrusion detection research, as surveyed by Jones and Sielken (2000), provided an early comprehensive survey of intrusion detection, categorizing methods into misuse detection (for known attacks) and anomaly detection (for behavioral deviations). They highlighted that while misuse detection is effective for known threats, it struggles with novel attacks, a gap that anomaly detection aims to fill. They further emphasized the limitations of early systems regarding scalability and adaptability to evolving threats, stressing the need for effective trade-offs among performance metrics like accuracy and false positive rates.

Building on this foundation, Chandola et al. (2009) conducted a systematic survey, classifying anomaly detection into statistical, machine learning, and information-theoretic methods. This seminal work stressed the difficulty of defining an anomaly across diverse domains and detailed algorithms like k-means and GMM. A key conclusion was the challenge posed by high-dimensional data and the scarcity of labeled datasets, leading the authors to advocate for future research into unsupervised, scalable methods capable of handling dynamic environments and large data volumes in real-time.

Specific algorithm exploration demonstrated various performance trade-offs. Yu et al. (2023) applied Gaussian Mixture Models (GMM) for network intrusion, achieving an F1−score of 0.80 by modeling complex data distributions, but noted the limitation that real-world network traffic might not always follow a Gaussian assumption. Addressing the issue of limited labeled data, Santos et al. (2019) explored semi-supervised learning using clustering and distance-based scoring, reporting a recall rate of 75% and precision of 70%. Their work highlighted the potential of such approaches but acknowledged issues when managing high-dimensional data and real-time streams.

The use of hybrid models to improve detection was explored by Bai et al. (2020), who integrated Principal Component Analysis (PCA) for dimensionality reduction with the Random Forest (RF) algorithm. This hybrid approach achieved a high detection accuracy of 92% and an F1−score of 0.91 on the CICIDS dataset, showcasing its effectiveness for identifying known and novel attacks. However, the study conceded a key limitation, which is its reliance on labeled data restricted its ability to generalize to completely unseen attack types, suggesting a need for unsupervised or semi-supervised enhancements.

Deep learning and advanced sequential modeling emerged as high-accuracy contenders. Wu et al. (2020) explored Long Short-Term Memory (LSTM) networks for real-time detection in IIoT, achieving a remarkable accuracy rate of 94% by effectively capturing temporal dependencies. However, they noted a significant challenge in the model's high computational demands, resulting in latency that made it less suitable for time-sensitive, critical applications. This highlighted a gap between high accuracy and computational efficiency in deep learning models.

Similarly, Kim et al. (2021) investigated autoencoders for network anomaly detection, training them on normal traffic and using reconstruction error for anomaly identification. Their work reported a strong detection accuracy of 93%, demonstrating deep learning's power in capturing complex patterns. Yet, like LSTM, they acknowledged the computational intensity and extended training times of autoencoders, suggesting that optimization is needed to maintain accuracy while operating within real-time detection constraints.

Barbariol et al. (2021) provided a comprehensive overview of various tree-based methods—including Decision Trees, Random Forests, and advanced techniques like Isolation Forests—for anomaly detection. They detailed the strengths of these methods in handling non-linear data and their interpretability. However, they also addressed common limitations, such as sensitivity to hyperparameter tuning and potential overfitting, emphasizing the growing importance of integrating

ensemble and hybrid models for adaptability in dynamic environments.

The systematic review by Al Farizi et al. (2021) focused specifically on the Isolation Forest algorithm. They confirmed its strengths in efficiently detecting anomalies in high-dimensional and large-scale datasets due to its tree-based structure. The review noted its effectiveness in both supervised and unsupervised settings and its ability to handle noise. A crucial gap identified was the need for more research into the algorithm's application for real-time anomaly detection in emerging fields like cybersecurity and IoT, suggesting further work to fully harness its potential in highly dynamic environments.

Collectively, the literature highlights a tension between the high accuracy of supervised deep learning (like LSTM and autoencoders) and the computational efficiency and adaptability of methods like Isolation Forest and semi-supervised techniques. The consensus points toward the need for scalable, hybrid, and computationally optimized solutions that can effectively process massive, dynamic data streams and generalize to novel threats without relying solely on exhaustive labeled datasets.

III. METHODOLOGY

This study employed a comparative research design centered on controlled experiments to evaluate the effectiveness of three distinct machine learning techniques—Isolation Forest (IF), Long Short-Term Memory (LSTM), and Q-Learning—for cyberattack anomaly detection in network traffic. This systematic approach ensured the reproducibility and validity of the results by applying all algorithms to a standard dataset under consistent conditions. The experimental workflow was structured, encompassing crucial steps like data preprocessing, feature extraction, model training, and hyperparameter tuning. Performance was rigorously assessed using a standard set of metrics, including precision, recall, F1-score, False Positive Rate (FPR), False Negative Rate (FNR), and Accuracy, to determine the relative effectiveness of each technique for comparison.

The research system was conceptualized as a real-time anomaly detection system designed to process continuous network data flows. It comprised four main operational modules:

(i) Data Acquisition, which was responsible for continuously collecting data from sources like network logs and SIEM systems;
(ii) Data Preprocessing, which cleaned, filtered, normalized, and prepared the raw data; and
(iii) Feature Engineering, which extracted relevant attributes, including statistical, time-series, and domain-specific features.
(iv) The core component was the Streaming Machine Learning Module, which utilized the IF, LSTM, and Q-Learning algorithms to identify anomalies based on isolation levels and temporal patterns.
(v) This system was completed by the Alert Generation Module, which provided timely notifications to security analysts upon detection.

3.1. Dataset Acquisition

The CICIDS-2017 dataset was selected for this study as a widely recognized benchmark for network-based anomaly detection. It is a comprehensive, labeled dataset consisting of 2,830,743 total records, representing both normal traffic and 14 types of cyberattacks, including DDoS and brute force. With 77 features and 80 network characteristics like flow duration and packet size, the dataset's high dimensionality and diverse scenarios make it highly suitable for testing the adaptability and scalability of the selected supervised and unsupervised machine learning algorithms.

3.2. Data Preprocessing and Feature Engineering

Data preprocessing and feature engineering were foundational to this research, ensuring the raw CICIDS-2017 network traffic data was clean, informative, and standardized for the machine learning models. This phase comprised a systematic series of steps designed to maximize the predictive power and reliability of the subsequent anomaly detection process. The initial and most critical step was Data Cleaning, which systematically addressed imperfections in the dataset, focusing on handling missing values, managing outliers, standardizing data types, and reducing noise.

The data cleaning procedure handled missing values and outliers. Of the ≈65,000 records (2.3%) initially containing missing entries, the 20,000 records with heavy missingness (≥30% empty fields) were removed. Remaining missing numerical and categorical features were filled using the median and mode, respectively, to minimize skew. Outlier

© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

management, identified using Z-score analysis, saw 5,100 of the ≈8,300 outliers removed. The remaining 3,200 rare attack patterns were capped within acceptable thresholds to preserve meaningful information while preventing model skewing. The data cleaning procedures are summarized in Table 3.1.

Table 3.1
Summary of the Data Cleaning Procedures

Cleaning Procedure	Target	Quantity/ Action	Result / Technique
Missing Value Handling	Records with missing/ invalid values	≈ 65,000 records (2.3%) identified.	Records with ≥30% empty fields (20,000) were removed.
Imputation	Remaining missing numerical features.	N/A	Imputed using the median value (numerical) and mode (categorical) to minimize skew.
Outlier Management	Extreme outliers (detected via Z-score).	≈ 8,300 records (0.3%) identified.	5,100 outliers were removed; 3,200 (rare attack patterns) were capped (Winsorized).
Duplicate Removal	Redundant records causing overfitting.	12,000 records (0.4%) eliminated.	Used hash-based checking to ensure training sample uniqueness.
Feature Redundancy	Features causing multicollinearity.	15 features dropped.	Identified via correlation analysis (coefficient≥0.9).
Data Type Conversion	Categorical strings (e.g., protocol type).	N/A	Transformed using One-Hot Encoding (unordered) and Label Encoding (ordered).
Normalization	Continuous numerical features.	N/A	Applied Min-Max scaling to standardize feature scales, aiding deep learning stability.

Data standardization and noise reduction ensured machine readability and model stability. Categorical strings (e.g., protocol type) were converted using one-hot and label encoding, and continuous features were made uniform using floating-point formats. Noise reduction included the elimination of 12,000 duplicate records (0.4%) to prevent overfitting. Furthermore, correlation analysis identified 15 redundant features (coefficient≥0.9), which were dropped to reduce multicollinearity. Finally, Min-Max scaling was applied to continuous features, standardizing scales for stable deep learning model convergence.

The preprocessing phase concluded with Data Transformation and Normalization and Data Reduction. Transformation involved applying scaling techniques, such as Min-Max scaling, standardization, and robust scaling, to numerical features and encoding all categorical and text-based data into machine-readable numerical representations. Data Reduction was then performed, primarily using Principal Component Analysis (PCA), to simplify the dataset and reduce the curse of dimensionality, retaining essential structure while optimizing system performance and reducing computational overhead by selecting only the most relevant attributes.

The final phase, feature engineering, went beyond preparation to actively create valuable, informative features that enhanced model accuracy. This process involved: Feature Creation, where new statistical (e.g., mean, variance), time-series (e.g., seasonality), and domain-specific features were aggregated; Feature Transformation, where techniques like logarithmic and polynomial features were applied to capture non-linear relationships; and Feature Selection, which used correlation analysis, univariate selection, and Recursive Feature Elimination (RFE) to refine the feature set, ensuring only the most predictive attributes were used to train the final models.

3.3. Training of the Models

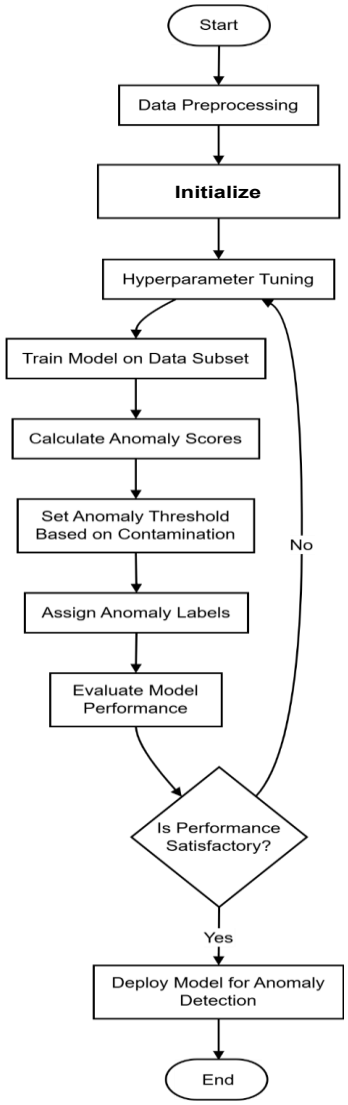
The model training and evaluation phase, illustrated with the flowchart in Figure 3.1, was vital for developing an accurate and reliable anomaly detection system, ensuring the suitability of the selected machine learning techniques for identifying cyberattacks in a live network setting. Despite differences in their underlying mechanisms, all three models—Isolation Forest, LSTM, and Q-Learning—underwent systematic training, fine-tuning, and evaluation using identical data splits and preprocessing steps. This uniformity ensured that the

© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

performance comparison was fair, reflecting the inherent capabilities of each model rather than external factors.

The Isolation Forest (IF), an unsupervised algorithm, was trained by modeling the data's structure to isolate anomalies without requiring labeled data. The process involved constructing multiple random decision trees on random subsets of the training data. Key hyperparameters, including the number of estimators (trees) and maximum samples, were tuned for efficiency and accuracy. The contamination rate, representing the expected proportion of anomalies, was a crucial parameter set based on the observed data imbalance. The trained IF model identified anomalies based on shorter average path lengths, which indicate instances that are easily isolated.

Figure 3.1
Designed Flowchart of Model Training and Evaluation Process



The Long Short-Term Memory (LSTM) network, a supervised deep learning approach, required labeled data to capture temporal dependencies. The training data was converted into sequential input-output pairs. The LSTM architecture, featuring hidden cells to mitigate the vanishing gradient problem, was trained using the Adam optimizer and the binary cross-entropy loss function. Hyperparameters like the number of LSTM units, batch size, and learning rate were optimized using grid search, while early stopping was employed to prevent overfitting and ensure the model's generalization to unseen data.

Q-Learning, a reinforcement learning algorithm, was trained by interacting with a network traffic environment defined by a state space and possible actions (normal or anomalous classification). A reward function was meticulously designed to maximize cumulative rewards by heavily penalizing both false positives and false negatives. Training involved an exploration phase to populate the Q-table, gradually shifting toward exploitation based on learned Q-values. Key parameters, including the learning rate (α), discount factor (γ), and exploration rate (ϵ), were tuned, with state-space discretization and ϵ-greedy exploration used to enhance efficiency in the high-dimensional network environment.

3.4. Model Evaluation

The study evaluated the effectiveness of the three selected machine learning techniques for anomaly detection using five key performance metrics (Precision, Recall, F1-Score, False Positive Rate, False Negative Rate, and Accuracy). This evaluation was necessary to provide comprehensive insight into each model's ability to accurately detect anomalies, with performance assessed on a separate test set to ensure reliability, particularly for unsupervised settings. The model training and evaluation flow chart is illustrated in Figure 3.1.

The formula for precision is:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)}+\text{False Positives (FP)}} \quad (3.1)$$

Mathematically, recall is defined as:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)}+\text{False Negatives (FN)}} \quad (3.2)$$

The F1 score is calculated as the harmonic mean of Precision and Recall:

$$\text{F1} = 2 * \frac{\text{Precision}*\text{Recall}}{\text{Precision}+\text{Recall}} \quad (3.3)$$

© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

False Positive Rate (FPR) formula is given as:

$$\text{FRP} = \frac{\text{False Positives (FP)}}{\text{False Positives (FP)+True Negatives (TN)}} \quad (3.4)$$

False Negative Rate (FNR) formula is given as:

$$\text{FRP} = \frac{\text{False Negatives (FN)}}{\text{False Negatives (FN)+True Positives (TN)}} \quad (3.5)$$

$$\text{Accuracy} = \frac{\text{Total Number of Predictions}}{\text{Number of Correct Predictions}} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (3.6)$$

3.5. Model Validation

Model validation was achieved using cross-validation, which splits the data into k-folds for robust stability and consistency testing. The comprehensive process involved preparing features and labels, initializing the Isolation Forest, LSTM, and Q-Learning models with specific parameters (like contamination rate), and finally assessing performance using metrics like Precision, Recall, F1-score, FPR, FNR, and Accuracy for a thorough evaluation.

IV. RESULTS

The comparative analysis of anomaly detection models yielded distinct performance outcomes, comprehensively evaluated using metrics including Accuracy, F1 Score, Precision, Recall, False Positive Rate (FPR), and False Negative Rate (FNR). The results established a clear hierarchy of efficacy across the three tested techniques, as detailed in Table 4.1.

Table
Performance Metrics of Anomaly Detection Models

Model	Isolation Forest	LSTM	Q-Learning
Accuracy	0.50774	0.99534	0.92802
F1 Score	0.46874	0.9934	0.90249
Precision	0.50848	0.99112	0.88618
Recall	0.51205	0.99573	0.92313
FPR	0.49586	0.00354	0.08585
FNR	0.48004	0.00499	0.06789

4.1. Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) model decisively outperformed the others, achieving nearly perfect detection capability. Its accuracy reached 99.53% with an F1 score of 0.993, demonstrating exceptional reliability. This superiority was confirmed by near-optimal precision (0.991) and recall (0.996), coupled with minimal error rates: a very low FPR of 0.004 and an FNR of 0.005. This success is primarily attributed to its supervised learning framework and inherent strength in capturing temporal dependencies within sequential network traffic data, aligning with findings in related studies (Wu et al., 2020).

4.2. Q-Learning

The Q-Learning model, representing the reinforcement learning approach, exhibited strong yet moderate performance, achieving an accuracy of 92.80% and an F1 score of 0.902. Its precision (0.886) and recall (0.923) were respectable but showed a tendency toward false negatives, indicated by a higher FNR of 0.068. The FPR of 0.086 pointed to occasional misclassification of normal traffic, a typical trade-off resulting from the algorithm's exploration-exploitation strategy in its adaptive learning process.

Q-Learning's adaptive policy optimization allows it to adjust dynamically to changing attack patterns, but its performance is limited by the discrete state-space representation and dependence on reward function design, which may not fully capture the complexity of network traffic features.

4.3. Isolation Forest

In stark contrast, the unsupervised Isolation Forest model performed poorly, demonstrating limited capability for this specific task with an accuracy of only 50.77%. Its F1 score of 0.469 signaled a poor balance between precision (0.508) and recall (0.512). The high error rates, FPR (0.496) and FNR (0.480), emphasized its failure to reliably distinguish between normal and anomalous cases, likely due to its difficulty handling complex, high-dimensional, and imbalanced labeled datasets like CICIDS-2017.

4.4. Performance Metrics' Bar Plots

The bar plots from Figure 4.1 to Figure 4 offer detailed visual comparison of the three anomaly detection models, Isolation Forest, LSTM, and Q-Learning, across four key performance metrics, which are Accuracy, Precision, F1-Score, and Recall. Each metric is shown as a separate bar, allowing for a direct comparison of the models' strengths and weaknesses.

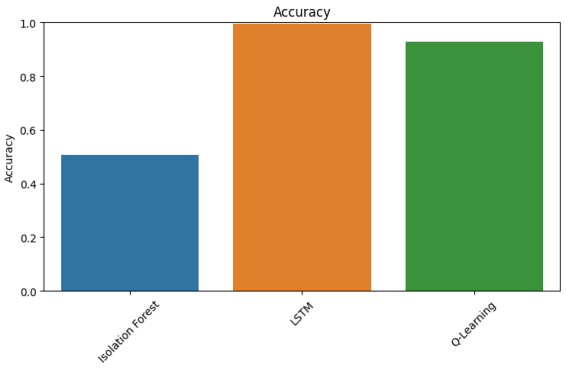
© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

Accuracy

Table 4.1 and the accuracy bar plot shown in Figure 4.1 show that the LSTM model demonstrated a superior performance, achieving a near-perfect Accuracy of 99.53%, the highest among the techniques. This result is directly attributed to its supervised learning approach, which effectively utilizes labeled data to establish precise decision boundaries. In contrast, Q-Learning showed a solid but lower accuracy of 92.80%, reflecting the trade-offs inherent in its adaptive reinforcement learning methodology. The unsupervised Isolation Forest performed significantly worse at 50.77%, consistent with its difficulty in handling complex, imbalanced datasets without the guidance of labeled examples.

Figure 4.1

Accuracy Bar Plot

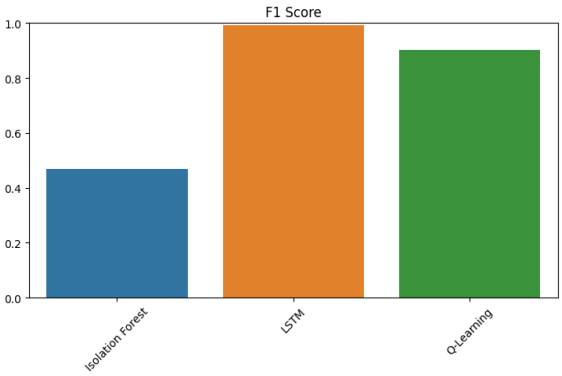


F1 Score

The F1-Score shown in Table 4.1 (LSTM: 99.34%, Q-Learning: 90.25%, Isolation Forest: 46.87%) and the visualized bar plot in Figure 4.2, successfully balance the models' precision and recall, confirming LSTM's strong robustness. This metric highlights Q-Learning's slight performance trade-off due to its adaptive policy and reveals Isolation Forest's inadequacy for high-stakes detection tasks. The high Recall (LSTM: 99.57%, Q-Learning: 92.31%) further confirms LSTM's crucial ability to detect nearly all true attacks, which is essential for critical systems, while Q-Learning's lower figure is likely a consequence of its exploration-exploitation dilemma during training.

Figure 4.2

F1 Score Bar Plot

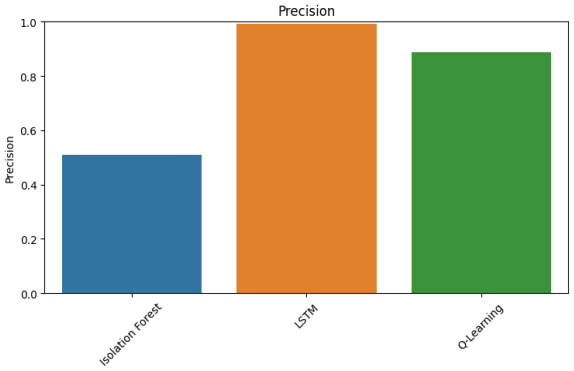


Precision

Precision, which measures the ratio of correctly identified anomalies to all predicted anomalies, is crucial for minimizing false alerts. As shown in Table 4.1 and the Precision Bar Plot, the LSTM model excelled with a 99.11% precision, indicating very few false positives. Q-Learning was respectable at 88.62% but showed a higher chance of misclassifying normal traffic. Conversely, Isolation Forest performed poorly, with only 50.85% precision, suggesting it was barely better than random chance at distinguishing true anomalies from noise in the complex CICIDS-2017 dataset.

Figure 4.3

Precision Bar Plot



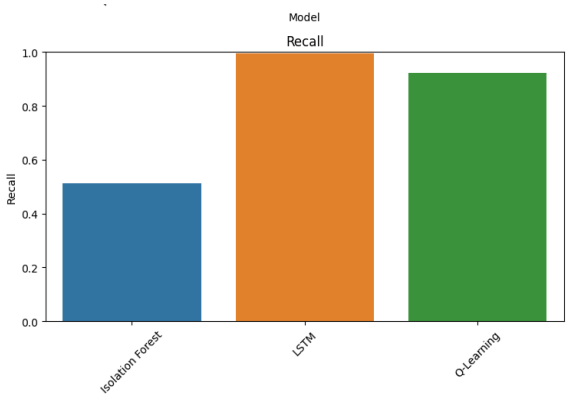
Recall

The Recall results, as summarized in Table 4.1 and illustrated in Figure 4.4, show that Isolation Forest's recall of 51.20% is insufficient for practical security, as it indicates the model would fail to detect nearly half of all actual attacks. This performance gap is causally linked to the models' underlying mechanisms: LSTM's superiority stems from its supervised training, which uses labeled data to minimize classification error directly, and its ability

© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

to model temporal dependencies in sequential traffic. Conversely, Q-Learning relies on indirect optimization via a reward function, and the Isolation Forest depends solely on unsupervised isolation paths, revealing the limitations of these methods on complex, labeled, and imbalanced datasets.

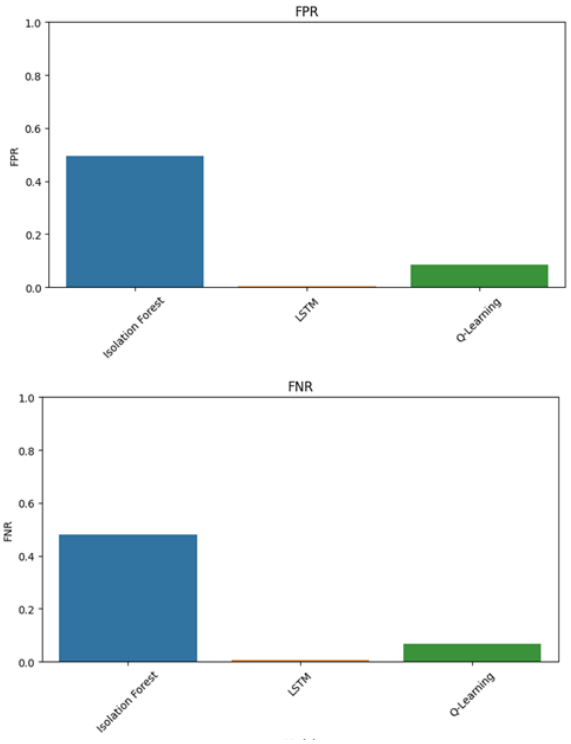
Figure 4.4
Recall Bar Plot



False Positive Rate (FPR) and False Negative Rate (FNR)

The analysis of error rates, visualized in Figure 4.2, highlights critical performance trade-offs for both false alarms and missed attacks. The Isolation Forest exhibited the highest FPR (49.59%) and a high FNR (48.00%), meaning nearly half of both its alerts and its missed attacks were errors, thus undermining trust due to its unsupervised nature. Conversely, the LSTM model achieved a near-zero FPR (0.35%) and a minimal FNR (0.50%), demonstrating superior accuracy in distinguishing genuine threats, a benefit of its supervised training on labeled sequences. Q-Learning presented a moderate FPR (8.58%) and a moderate FNR (6.79%), reflecting occasional misclassifications during its policy exploration phase. This hierarchy confirms that model selection requires balancing the need for low error rates with the available labeling infrastructure and computational resources.

Figure 4.5
False Positive Rate and False Negative Rate Comparison plot



V. DISCUSSION OF RESULTS

The results strongly align with and extend existing anomaly detection research. LSTM's dominance (99.53% accuracy) confirms findings by Kim et al. (2021) that deep learning excels in network analysis, processing sequential data hierarchically to capture dependencies. In contrast, Isolation Forest's poor performance reflects the challenge noted by Barbariol et al. (2021), that static tree-based methods struggle with high-dimensional, imbalanced data without the benefit of explicit labeled guidance or sophisticated temporal modeling.

Q-Learning's intermediate performance highlights the difficulty of applying reinforcement learning in this domain. While its adaptive policy is theoretically beneficial for dynamic environments (Sutton & Barto, 2022), practical issues like state-space complexity are evident. The observed FNR (6.79%) suggests it occasionally misses attacks, a limitation also seen in semi-supervised systems (Santos et al., 2019). However, its lower FPR (8.58%) compared to Isolation Forest shows good specificity, which is valuable for minimizing security alert fatigue.

The comparative analysis establishes a clear hierarchy of performance: LSTM>Q-

© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

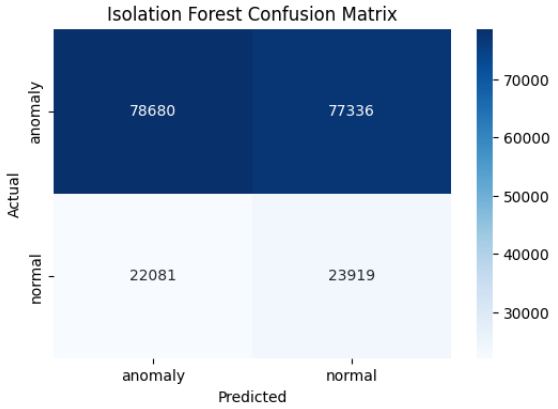
Learning>Isolation Forest. This aligns with the literature (Bai et al., 2020; Chandola et al., 2009), favoring supervised and adaptive models over unsupervised methods on labeled data. The selection ultimately depends on operational constraints: LSTM is ideal for high-stakes environments where accuracy is paramount, Q-Learning suits dynamic systems needing adaptability, and Isolation Forest remains practical only for computationally light, initial anomaly screening. Figure 4.5 visually emphasizes these trade-offs. LSTM clusters near the optimal origin, while Isolation Forest is constrained by high error rates. Q-Learning sits in the middle, indicating its utility when a moderate error rate is acceptable for adaptability. This confirms that model choice is context-dependent. Future research should explore hybrid models, combining LSTM's accuracy with Q-Learning's adaptability, to further improve real-world cybersecurity performance.

The Confusion Matrix for Isolation Forest (Figure 4.6) provides the visual proof for its inadequacy. The off-diagonals show massive and nearly equal misclassification rates: 77,336 normal instances incorrectly flagged as anomalies (False Positives) and 22,081 true anomalies missed (False Negatives). These immense error counts, which directly explain the model's high FPR and FNR, demonstrate that the unsupervised method failed to establish a meaningful separation boundary in the complex feature space. This fundamental lack of discriminative power is further confirmed by its ROC Curve (Figure 4.7). The curve barely deviates from the diagonal dashed line, which represents random guessing. The AUC value clings closely to 0.50, solidifying the conclusion that Isolation Forest is an ineffective tool for this specific supervised anomaly detection task, despite its proven utility in general, purely unsupervised outlier detection scenarios.

In stark contrast, the LSTM's Confusion Matrix (Figure 4.8) is the model of a high-performing classifier. The matrix shows the vast majority of instances correctly identified (45,574 true anomalies and 155,935 true normal instances), with errors being minuscule: only 426 false positives and 81 false negatives. This visual evidence of precision and reliability is directly attributable to the model's supervised training on sequential data, allowing it to learn the intricate temporal patterns that define attacks. This flawless performance is captured by its ROC Curve (Figure 4.9), which shoots to the top-left corner, achieving a perfect AUC of 1.00.

Figure 4.6

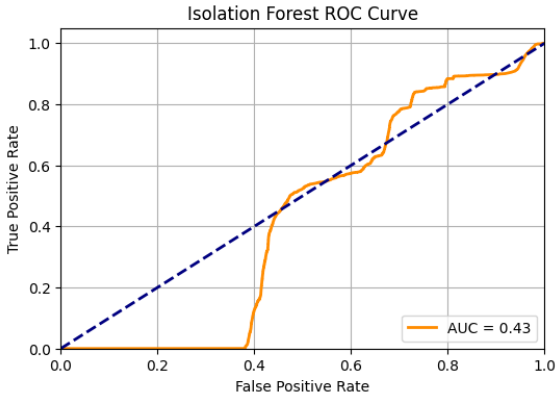
Confusion Matrix for the Isolation Forest model.



Note. Confusion Matrix for the Isolation Forest model. The high values in the off-diagonal cells (77,336 and 22,081) visually represent its high rate of misclassifications.

Figure 4.7

Receiver Operating Characteristic (ROC) curve for the Isolation Forest model.

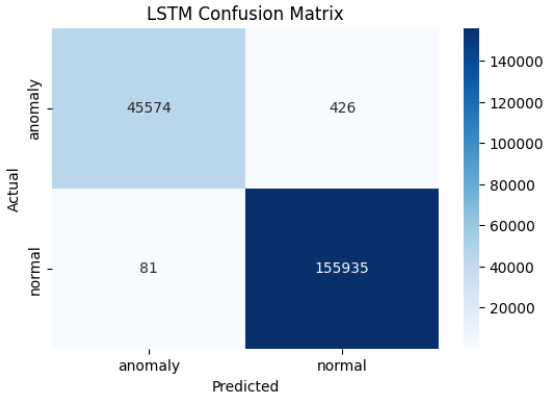


Note. Receiver Operating Characteristic (ROC) curve for the Isolation Forest model. The curve tracking close to the diagonal (AUC ≈ 0.50) indicates performance no better than random chance.

The Q-Learning Confusion Matrix (Figure 4.10) shows a capable, adaptive model with visible errors: 4,140 false negatives (missed attacks) and 4,017 false positives. This error pattern is typical of reinforcement learning's exploration process. The model's ROC Curve (Figure 4.11) confirms very good performance, showing a strong climb away from random guessing, yet it falls short of the LSTM's perfection. This indicates Q-Learning successfully finds a highly effective decision boundary but entails a slight sensitivity-specificity trade-off.

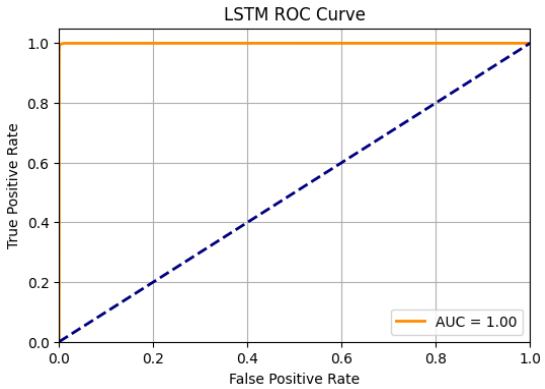
© OCT 2025 | IRE Journals | Volume 9 Issue 4 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I4-1711113-3371

Figure 4.8
Confusion Matrix for the LSTM model.



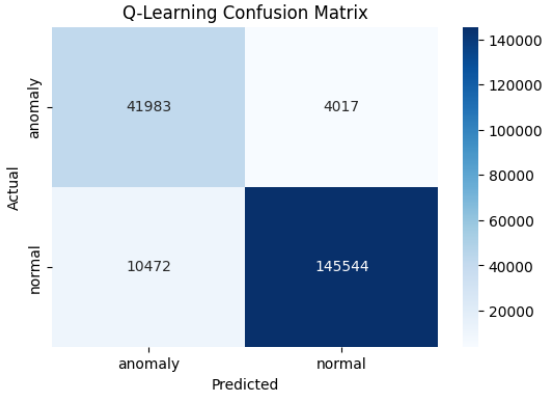
Note. Confusion Matrix for the LSTM model. The overwhelming values on the main diagonal (45,574 and 155,935) and minimal off-diagonal values (426 and 81) demonstrate its exceptional accuracy.

Figure 9
ROC curve for the LSTM model.



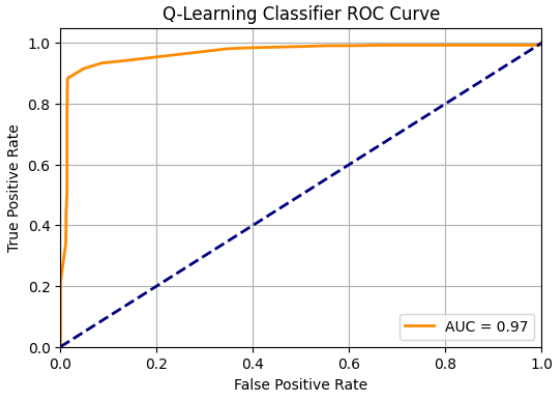
Note. ROC curve for the LSTM model. The curve achieving an AUC of 1.00 indicates perfect classification performance across all thresholds.

Figure 4.10
Confusion Matrix for the Q-Learning model.



Note. Confusion Matrix for the Q-Learning model. It shows strong correct classification counts (41,983 and 145,544) but with noticeable error rates (4,140 and 4,017), indicating its adaptive but imperfect nature.

Figure 11
ROC curve for the Q-Learning model.



Note. ROC curve for the Q-Learning model. The curve shows strong performance by arching towards the top-left corner, though not perfectly, indicating a good but not optimal trade-off between TPR and FPR.

Summary of Findings from Evaluation Metrics

1. Overall Performance and Accuracy

- LSTM Dominance: The Long Short-Term Memory (LSTM) model achieved unequivocally superior performance, with a near-perfect Accuracy of 99.53% and the highest F1 Score (0.993), confirming its capability for reliable cyberattack anomaly detection in this setting.
- Q-Learning Viability: The Q-Learning model showed strong, yet moderate, performance, achieving an Accuracy of 92.80% and an F1 Score of 0.902, positioning it as a viable, adaptable alternative.
- Isolation Forest Failure: The Isolation Forest model performed poorly, with an Accuracy of only 50.77% and an F1 Score of 0.469, indicating it was little better than random chance for this specific complex, labeled task.

2. Precision and Recall (Reliability and Coverage)

- Precision (Minimizing False Alarms): LSTM exhibited exceptional reliability with Precision at 99.11%, ensuring very few false positives. Q-Learning was respectable at 88.62%, while Isolation Forest's 50.85% precision was insufficient for practical use.
- Recall (Minimizing Missed Attacks): LSTM also had the highest coverage, with Recall at 99.57%, meaning it successfully detected the vast majority of attacks. Q-Learning achieved a

high recall of 92.31%. Isolation Forest's recall was only 51.20%, meaning it would fail to detect nearly half of all true attacks.

3. Error Rates (FPR and FNR)

- Optimal Error Rates (LSTM): LSTM demonstrated minimal errors in both categories: False Positive Rate (FPR) of 0.004 and False Negative Rate (FNR) of 0.005.
- Acceptable Trade-Off (Q-Learning): Q-Learning showed a moderate trade-off, with an FPR of 0.086 and an FNR of 0.068, reflecting its adaptive, policy-based learning.
- Unacceptable Error Rates (Isolation Forest): Isolation Forest showed unacceptable error rates, with a high FPR of 0.496 and a high FNR of 0.480, confirming its inability to set meaningful detection boundaries.

4. Causal Factors

- The superior performance of LSTM is directly attributed to its supervised learning framework and its ability to model temporal dependencies in sequential network traffic data.
- Q-Learning's intermediate performance reflects its reliance on indirect optimization via a reward function and its exploration-exploitation strategy.
- Isolation Forest's failure is linked to the limitations of unsupervised methods when applied to complex, labeled, and imbalanced cybersecurity datasets.

VI. CONCLUSION AND RECOMMENDATIONS

This study conducted a comparative analysis of three distinct machine learning paradigms—Isolation Forest (IF), Long Short-Term Memory (LSTM), and Q-Learning—for cyberattack anomaly detection, implementing the system using the CICIDS-2017 dataset in Python. The evaluation, based on comprehensive metrics including Accuracy, F1-score, FPR, and FNR, decisively established the LSTM model as the superior performer in terms of detection efficacy. Operating within a supervised framework, the LSTM network achieved a near-perfect Accuracy of 99.53% and an excellent F1−score of 0.993, with a flawless Area Under the Curve (AUC) of 1.00. Crucially, its ability to minimize errors was demonstrated by a minimal False Positive Rate (FPR) of 0.35% and a False Negative Rate (FNR) of 0.50%.

The findings confirm that sophisticated supervised deep learning is the optimal choice for environments with abundant labeled historical data where maximum detection accuracy is the primary objective. The Q-Learning model, representing the reinforcement learning approach, presented a valuable alternative, showcasing strong capability with an Accuracy of 92.80% and an F1−Score of 0.902. While adaptive, its trade-offs were evident in its higher error rates compared to LSTM (FPR of 8.58% and FNR of 6.79%). This indicates that Q-Learning holds significant potential for continuous improvement in dynamic, evolving threat environments where adaptability to novel attacks outweighs the need for static, maximum-level accuracy.

In stark contrast, the unsupervised Isolation Forest model proved fundamentally inadequate for this specific, supervised anomaly detection task. Its performance metrics, which clustered around 50%, demonstrated efficacy no better than random chance. This leads to the conclusive finding that simple unsupervised techniques, in their standalone form, are not a viable solution for targeted cyberattack anomaly detection on complex, labeled cybersecurity datasets. The research thus provides a data-driven framework, confirming the dominance of LSTM, establishing the practical promise of adaptive Q-Learning, and clearly delineating the limitations of the Isolation Forest unsupervised method.

RECOMMENDATIONS

Based on these findings, the Long Short-Term Memory (LSTM) model is the primary recommendation for operational security where maximum detection accuracy is paramount (e.g., critical infrastructure). Its superior performance justifies the investment in labeled datasets and hardware (GPUs/TPUs) to minimize computational overhead. Conversely, Q-Learning is recommended for dynamic environments with evolving threats, where organizations should explore Deep Q-Learning Networks (DQNs) and meticulously design the reward function. This approach prioritizes adaptability to zero-day attacks over static, near-perfect accuracy.

Future research should focus on hybrid intelligent systems that synergize the models' strengths. The most promising path is a hybrid LSTM-Q-Learning architecture, where LSTM provides sophisticated,

temporally-aware feature extraction for the Q-Learning agent. Efforts should also refine Q-Learning through multi-agent reinforcement learning (MARL). Simple unsupervised methods like Isolation Forest should be repurposed solely for initial, lightweight filtering within a tiered detection architecture, rather than serving as a standalone solution.

REFERENCES

[1] Al Farizi, W. S., Hidayah, I., & Rizal, M. N. (2021). Isolation Forest Based Anomaly Detection: A Systematic Literature Review. *2021 8th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE)*, 118–122. https://doi.org/10.1109/ICITACEE53184.2021.9617498

[2] Al-Shaymaa, Y., Yassine, A., & Hajjdi, A. (2019). Anomaly detection in distributed systems: A survey of the state-of-the-art. *IEEE Communications Surveys & Tutorials*, *21*(3), 2250–2283.

[3] Bai, X., Zhang, Y., Wang, S., & He, H. (2020). Network anomaly detection using a hybrid method of random forest and principal component analysis. *IEEE Access*, *8*, 79415–79427. https://doi.org/10.1109/ACCESS.2020.2990101

[4] Barbariol, T., Chiara, F. D., Marcato, D., & Susto, G. A. (2021). A Review of Tree-Based Approaches for Anomaly Detection. *Control Charts and Machine Learning for Anomaly Detection in Manufacturing*, 149–185. https://doi.org/10.1007/978-3-030-83819-5_7

[5] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys (CSUR). *Springer*, *41*, 58. https://doi.org/10.1145/1541880.1541882

[6] Clarke, R., Stavrou, L., & Wright, G. (2018). The Net: A Tailored Cyber Threat Intelligence Framework for Cyber Security. *Journal of Information Security*, *9*(01), 1–17.

[7] Jones, S. B., & Sielken, R. S. (2000). Computer system intrusion detection: A survey. *University of Virginia Technical Report*, *35*, 16–22.

[8] Kim, M., Lee, C., & Kim, J. (2021). Anomaly detection using autoencoders in network traffic. *IEEE Transactions on Information Forensics and Security*, *16*(2), 320–328. https://doi.org/10.1109/TIFS.2020.3025136

[9] Kshetri, N. (2017). Cybersecurity and cyberwarfare: A review of the literature. *Journal of Information Technology & Politics*, *14*(2), 149–164.

[10] Modi, C., Patel, D., & Borisaniya, B. (2017). A survey on anomaly detection in network traffic. *Journal of Intelligent Information Systems*, *49*(2), 267–293.

[11] Santos, I., Brezo, F., & Ugarte-Pedrero, X. (2019). Semi-supervised anomaly detection for cybersecurity. *Journal of Information Security and Applications*, *44*, 146–156. https://doi.org/10.1016/j.jisa.2018.10.005

[12] Sutton, R., & Barto, A. (2022). *Reinforcement Learning: An Introduction* (2nd ed.). MIT Press.

[13] Wu, Z., Liu, J., & Zhang, J. (2020). Real-time anomaly detection using LSTM in industrial IoT networks. *IEEE Internet of Things Journal*, *7*(8), 7262–7270. https://doi.org/10.1109/JIOT.2020.2972211

[14] Yu, B., Zhang, Y., Xie, W., Zuo, W., Zhao, Y., & Wei, Y. (2023). A Network Traffic Anomaly Detection Method Based on Gaussian Mixture Model. *Electronics*, *12*(6), 1397. https://doi.org/10.3390/electronics12061397