IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 13 Issue 06, June 2026
ISSN (Online) 2348 – 7968 | Impact Factor – 6.72
www.ijiset.com

Comparative Assessment of Common Machine Learning Techniques for Network Intrusion Detection Using the CIC-IDS2017 Dataset

Rania Elsheikh Hamid Ibrahim
Management information systems, Sudan University of Science and Technology/,
Khartoum, State 11111, Sudan

Abstract
This paper presents a thorough empirical evaluation of four classical machine learning algorithms, namely Logistic Regression, Support Vector Machine (SVM), Random Forest, and Extreme Gradient Boosting (XGBoost), using the benchmark CIC-IDS2017 dataset. To address the specific problems of the data such as infinite values, unbalanced classes, an accurate planning methodology was used implementing the SMOTE technique. The results show that the XGBoost algorithm outperformed the other models, with the highest accuracy of 93.67% and a ROC-AUC value of 98.13%. Moreover, the algorithm demonstrated a short training time of 0.59 seconds and high operational efficiency. Following the implementation of the Explainable AI (XAI) structure—more precisely, SHAP—to address the "black-box" problem, it became evident that features such as flaw duration had the greatest impact on detection decisions. To offer a practical and deployable solution for real Security Operations Centres (SOCs), this research article develops a balanced framework that integrates high accuracy, processing efficiency, and clarity.

Keywords: *Network intrusion detection, CIC-IDS2017 Dataset, Explainable AI(XAI), SHAP Algorithm.*

1. Introduction

The frequency and complexity of cyberattacks have grown significantly in the current digital era, threatening an organization's fundamental infrastructure and information security. Traditional signature-based Intrusion Detection Systems (IDS) are not effective on their own due to the advancement of attacker methods, particularly the rise of Advanced Persistent Threats (APTs) and "zero-day" attacks. These types of systems cannot detect new patterns of actions or attacks where signatures are hidden since they rely on regularly revised databases of previous attacks [1].

Machine learning (ML), that can learn the behavioural patterns of regular network traffic and reliably detect unusual activity, is now recognized as a promising substitute for anomaly-based detection systems in attempt to navigate around such limitations. Still, the quality and accuracy of the dataset used for training and testing are essential to any machine learning model's efficacy. Several studies in previous years relied on outdated datasets such KDD99 or NSL-KDD, which received severe criticism for having a significant class imbalance, absence of the features characteristic of modern network traffic, and exhibiting unrealistic attack scenarios.

The CIC-IDS2017 dataset, constructed by the Canadian Institute for Cybersecurity (CIC), has been considered as the most current and dependable benchmark in this domain. In addition to providing eight modern attack types, such as Brute-force, Denial of Service (DoS/DDoS), Web Attacks, Infiltration, and Botnet, it provides complete, pre-generated features while simulating a realistic network environment over five days [2].

Using the CIC-IDS2017 dataset, this paper aims to provide an accurate comparative empirical evaluation of four common traditional based machine learning algorithms in the field of network intrusion detection. The paper's main contributions are:
1. Applying a methodical standard preparation approach to address the difficulties of the CIC-IDS2017 dataset, such as missing Nan and infinite Infinity values.
2. Executing a comprehensive performance comparison using several measurement metrics, keeping special attention to the compromises between inference speed (Training Time) and accuracy.
3. Implementing. Explainable AI (XAI) algorithms, like the SHAP algorithm, to provide comprehensible insights into the most important characteristics in detection decisions, improving SOC analysts' trust as well as making result reviews simple.

2.Relatedwork

During the last ten years, there has been a significant amount of interest in research around the implementation of ML approaches in IDS. Three main dimensions may be used to categorize previous, related work:

First: Evolution and Comparison of Datasets

The basic shortcomings of traditional datasets were pointed out by Khan et al. (2024), who noted that they do not accurately reflect the features of network traffic[1] . On the other hand, while CIC-IDS2017 was created to capture actual attack incidents and contains a considerable amount of regular traffic, Sharafaldin et al. (2018) verified that it demonstrates a quantum advancement [2]. However, research warned about problems with this dataset's quality of the data, such as a considerable class imbalance in certain attacks (such Heartbleed or Infiltration), a requirement for rigorously preprocessing.

Second: Comparing Machine Learning Algorithms for Intrusion Detection

ML algorithms were implemented in numerous studies for evaluating CIC-IDS2017. When Azizan et al. (2021) compared SVM with Random Forest (RF), for example, they observed that RF were having the highest recall percentage and SVM were having an accuracy of 98.18% [3]. Moreover, Thapa et al. (2020) showed that traditional tree-based algorithms (such as CART) achieve the optimum level of balance between minimal training time and high accuracy (>99%), completely higher than computationally pricey Deep Learning models[4].

Waghmode et al. (2025) proposed implementing an IDS framework employing a Quantum-inspired Least Square Support Vector Machine (LS-SVM) combined with Exhaustive Feature Selection (EFS) to solve the computational constraints of traditional Support Vector Machines (SVMs). At an unusually fast testing time of 1.0 seconds and a 99.5% accuracy on the CIC-IDS-2017 dataset, the technique they used showed that it was highly efficient in minimizing the high false-positive rates usual of ML-based IDS. Still, with their LS-SVM model's improved computational speed and

forecasting accuracy, it is considered as a "black-box," supplying no post-hoc justification for its detection results[5].

Third: Cybersecurity and Explainable AI (XAI)

Although ML models' efficacy, acceptance in SOCs is severely constrained by their "black-box" nature. To increase clarity, attention recently changed to implementing XAI. As stated by Mohale and Obagbuwa (2025), adopting SHAP in combination with models like XGBoost strengthens analysts' capability to classify warnings while also assisting in the detection of data anomalies as well as improving system stability[6].

Research Gap:

Although the reality that CIC-IDS2017 was used in several research studies, many of them either skip the interpretability factor or emphasis on improving accuracy using complex models (such as Hybrid Deep Learning, which produces high accuracy but at a substantial computational expense and long inference period [7]). With the goal to bridge this gap, this study offers a balanced framework that consists of: (1) methodical preprocessing for real-world data challenges; (2) a transparent comparison of the performance and efficiency of four ML algorithms; and (3) implementation of XAI (SHAP) to interpret the decisions generated by the ideal model.

3. Methodology

The experimental framework for investigating the ML algorithms is presented in this section.

3.1 Description of the Dataset

The CIC-IDS2017 dataset was employed in this research. Around 11.8 million instances and 85 attributes that were extracted previously from network flows (NetFlow) have been included in the data, which was gathered over five days in eight various sessions. Attacks such as brute-force, DoS/DDoS, web attacks, infiltration, and botnets are contained.

3.2 Preparing Data

To guarantee model stability, CIC-IDS2017 introduces technical issues that should be addressed precisely:

1. Addressing Anomalies and Missing Values: Mistakes in feature extraction occurred in the raw dataset's Infinity and Nan values. To prevent a potential algorithmic failure, these were exchanged out for the appropriate column's median.
2. Categorical Encoding: One-Hot Encoding was the method used to turn text-based variables into numerical forms.
3. Feature Scaling: Standard Scaling was applied to regularize feature ranges, which quickens up computation for algorithms for optimization and is necessary for distance-based algorithms like SVM.
4. Handling Class Imbalance: This study used the SMOTE (Synthetic Minority Over-sampling Technique) to produce synthetic samples for minority classes, guaranteeing that the model learns

minority patterns without discarding majority data, compared to some studies that used under sampling, which potentially results in a loss of crucial data from the Benign class[8].

3.3 A Few Machine Learning Techniques

To conduct an accurate comparison, four algorithms were selected as follows:
1. Logistic Regression (LR): Based on its speed and interpretability, LR is used as a theoretical baseline.
2. Support Vector Machine (SVM): Adopted due to its capacity to detect the ideal hyperplanes in high-dimensional spaces.
3. Random Forest (RF): A combination of ensemble technique that performs well with non-linear data and is particularly resistant to normal overfitting.
4. XGBoost: The revolutionary approach for structured tabular data, recognized for its regularization processes that reduce overfitting, integrated handling of missing values, and computing efficiency[8]

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \quad (1)$$

3.5 Assessment Measures and Experimental Configuration

It is misleading to rely only on accuracy due to the data's imbalance. As therefore, we employed Training Time as a significant operational measure in addition to accuracy, Recall, F1-Score, and ROC-AUC. Using a conventional stratified split, the data divided into 20% for testing and 80% for training. Python was implemented for these experiments (`scikit-learn, xgboost, imbalanced-learn, and Shap).

4. Results and discussion

Experimental results from the implementation of the four specified machine learning algorithms Logistic Regression, SVM, Random Forest, and XGBoost on the preliminary processed and SMOTE-balanced CIC-IDS2017 dataset are discussed and analyzed in this part. General performance of classifications, confusion matrix evaluation, operational efficiency, explainability, and benchmarking against the latest research are some of the features that make up the analysis.

4.1 The Total Performance in Classification

The Python programming language was used for all experimental implementations to provide a thorough and repeatable comparative examination of the chosen machine learning methods. A strong framework of specialized libraries was used to build the computational pipeline: the Extreme Gradient Boosting algorithm was implemented using the xgboost library, and the Logistic Regression, Support Vector Machine (SVM), and Random Forest models

Table 1: shows selected machine learning algorithms' cost of computation and complete performance metrics

Model	***Accuracy***	***Precision***	***Recall***	***F1-Score***	***ROC-AUC***	***Training Time (s)***
Logistic	82.90%	55.58%	77.10%	64.60 %	89.86%	0.17
SVM	83.23%	56.27%	76.94%	65.00%	89.87%	110.47
Random	93.67%	88.60 %	83.20%	85.81%	97.69%	4.36
XGBoost	94.43%	79.00 %	93.57%	85.67%	98.13%	0.53

The results presented show that tree-based machine learning algorithms—XGBoost and Random Forest in specific—performed notably more effectively than linear models. Using a ROC-AUC of 98.13%, XGBoost obtained a maximum accuracy of 94.43%. Logistic regression, on the opposite hand, showed the smallest average performance, which is in accordance with predicted results. The network's data in CIC-IDS2017 has complicated nonlinear interactions that are challenging for simple logical linear models to successfully explain. The above result significantly confirms recent studies conducted by Coşar et al. (2024) [8], resulting in found that boosting-based methods do better than linear approaches in extremely complex network traffic datasets (obtaining 98% accuracy and 1.00 AUC). Moreover, our findings compare well with the initial accuracy provided by Azizan et al. (2021)[3], where SVM reached 98.18% on the same dataset.

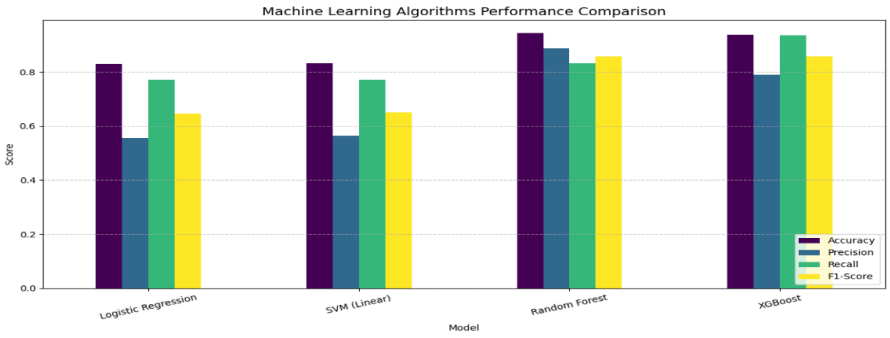

Fig. 1 illustrates the comparative performance metrics

4.2 Analysis of Confusion Matrix

The confusion matrix for the highest model (XGBoost) was analyzed to fully evaluate model behavior (Figure 2). decreasing False Negatives (FN), that signal a failure to recognize an actual attack and can allow lateral motion, is essential in cybersecurity. XGBoost managed to effectively reduce the FN rate to 39 entirely, due to the matrix. This advantage can be explained by XGBoost's capability to deal with class imbalance when coupled with SMOTE, enabling it to discover the complex topological behaviors of a minority attack categories without facing the information loss that occurs with under-sampling methodologies.

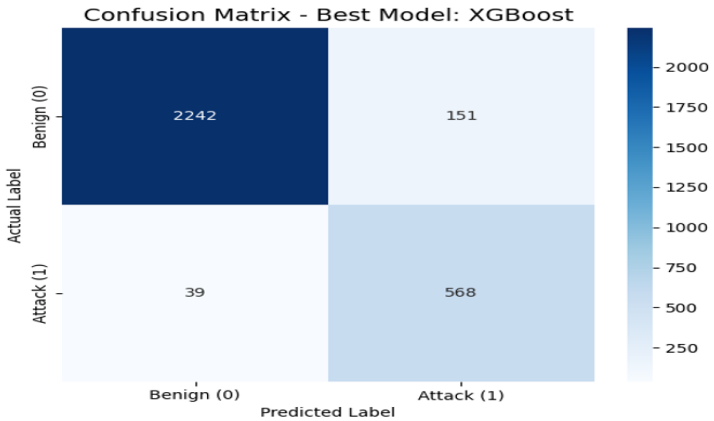

Fig. 2 Confusion Matrix Model XGBoost

4.3 Training Time and Operational Efficiency

This study evaluates training time as a key indicator of real-time practicality in addition to accuracy. Table 1 demonstrates that SVM took 110.47 seconds, while XGBoost required 0.53 seconds. The results obtained are in full agreement with Thapa et al. (2020), who showed that traditional machine learning approaches deliver the ideal potential balance between the high accuracy and extremely fast speed, greatly compared with Deep Learning models that might need hours or days to train [4]. Due to this, XGBoost and RF are optimal for IDS situations that need immediate responses and going on training.

4.4 SHAP Explainability Analysis

The "black-box" issue was addressed by integrating SHAP into the XGBoost model. The most essential factor in intrusion detection assessments was, Bwd Packet Length Max, followed by Fws IAT Total Packets, as shown to the SHAP Summary Plot (Figure 3). More specifically, there existed significant relationship between attack classification and higher values of Flow Duration (shown as red points on the positive SHAP axis), specifically for DoS attacks that are recognized by extended connection periods.

The implementation limitations mentioned by Mohale and Obagbuwa [6] are instantly solved by the incorporation of SHAP in our framework, specifically the decrease of "alert fatigue." Without having any deep knowledge of machine learning, SOC analysts can instantly assess incidents by transforming raw model outputs into human-readable feature attributions. Additionally, integrating SHAP to XGBoost makes it possible for SOC analysts to rely on high-accuracy the ensemble models through the provision of mathematically rigorous (Shapley values) local reasoning for each alert, thus bridging the gap between predictive performance and human comprehension, in accordance alongside the idea of "trust management" in cybersecurity highlighted by Mahbooba et al. (2021)[9].

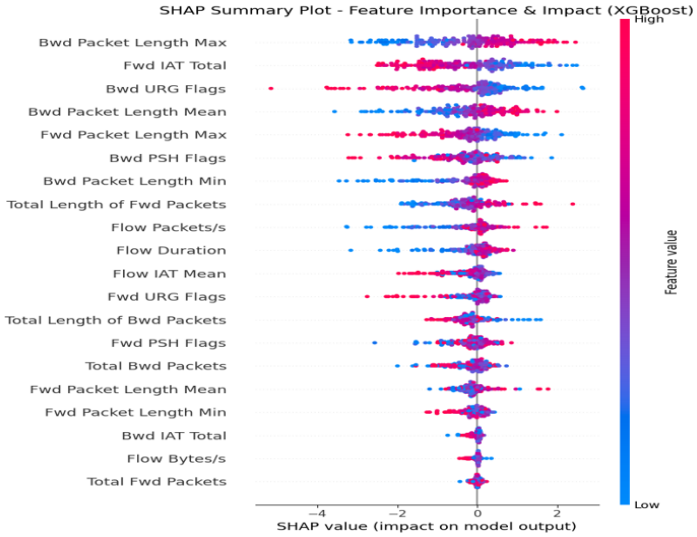

Fig. 3 SHAP summary plot illustrating the global feature importance and the direction of impact matrix.

4.5 Assessment of State-of-the-Art

This section presents scientific comparison of the implied XGBoost-based framework versus existing state-of-the-art techniques to integrate the results of this study into the larger context of current intrusion detection research. ML-driven intrusion detection systems (IDS) have become increasingly common due to the fast development of cyber-attacks. These frameworks highlight various elements of performance, such as model interpretability, computational efficiency, and forecast accuracy. However, the capacity to optimize for one aspect at the expense of others is an ongoing problem across many recent studies, as the literature review shows. Table 2 compares our method with current research to verify the reliability of the results.

Table 2: Benchmarks Compared with Current Research.

Study	**Algorithm Used**	**Dataset**	**Accuracy**	**XAI**
Azizan et al. (2021) [3]	SVM, RF, DJ	CIC-IDS2017	98.18% (SVM)	None
Thapa et al. (2020) [4]	CART, CNN+Emb	CIDDS / CIC-IDS2017	99.8% (CART)	Intrinsic (Feature Imp.)
Coşar et al. (2024) [6]	XGBoost, LightGBM	CSE-CIC-IDS2018	98.00%	None
Gaspar et al. (2024)[10]	MLP (Black-box)	ADFA-LD (IoT)	93.70%	Full (LIME vs. SHAP)
Mahbooba et al. (2021) [9]	Decision Tree (Intrinsic)	KDD Cup 99	98%	Intrinsic Rule Extraction
Waghmode et al. (2025) [5]	LS-SVM + EFS	CIC-IDS-2017, NSL-KDD, UNSW_NB1	99.50%	None
Proposed Methodology	XGBoost + SMOTE + SHAP	CIC-IDS2017	94.43%	Full (SHAP)

The evaluation indicates this approach not only obtained a better accuracy but also executes remarkably well by including full interpretability and measuring operational efficiency two metrics that past studies either missed or did not completely handle.

4.6. Discussion and Restrictions

Moreover, the proposed XGBoost based pipeline shows equivalent reasoning time comparable to state-of-the-art lightweight models in terms of its operational feasibility. For instance, our XGBoost model has the same real-time classified performance, but Waghmode et al. (2025) also obtained a very short testing time of 1.0 s while using LS-SVM on CIC-IDS-2017. Our structure improves this rate with SHAP-based explanations, compared to their black-box LS-SVM, delivering valuable knowledge to SOC analysts without any great deal of delay. Restrictions & Pipeline Projects: Types of data: The study used historical (offline) data (CIC-IDS2017). In real-life situations, attack patterns are constantly changing.

Perturbation Analysis: Future work will employ a “perturbation analysis” framework for network flow data, derived guidelines from the confirmation methodology of Gaspar et al. (2024)[8]. We intend to mathematically prove the robustness of the stated explanations by systematically varying the top-ranked SHAP features and observing the category variations.

Interactive Dashboards: Future work consists of an interactive XAI-enabled dashboard that automatically ranks the alerts based on SHAP-derived confidence scores, and leverages human-in-the-loop (HITL) feedback to continually enhance feature importance weights. This depends on the

evaluation framework founded on the Security Operations Center (SOC) established by Mohale and Obagbuwa (2025)[6].

5. Conclusion

There will be a requirement for advanced precision, efficient, and accessible Intrusion Detection Systems (IDS) because of the rapidly increasing complexity and variety of cyber-attacks. To develop a thorough Explainable AI (XAI)-based framework for comparing the performance of common algorithms for machine learning such as Logistic Regression, SVM, Random Forest, and XGBoost, the benchmark CIC-IDS2017 dataset was implemented. The suggested framework attempts to bridge the critical gap between high predictive accuracy and operational integrity in the existing cybersecurity environment using advanced specialized data preprocessing techniques, e.g. SMOTE for class imbalance management and standardized scaling, with post-hoc interpretability via SHAP.

References

[1] Z. I. Khan, M. M. Afzal, and K. N. Shamsi, 'A Comprehensive Study on CIC-IDS2017 Dataset for Intrusion Detection Systems', vol. 02, no. February, pp. 254–260, 2024.

[2] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, 'Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization', no. Cic, pp. 108–116, 2018, doi: 10.5220/0006639801080116.

[3] A. H. Azizan, S. A. Mostafa, A. Mustapha, C. Feresa, and M. Foozy, 'A Machine Learning Approach for Improving the Performance of Network Intrusion Detection Systems', vol. 5, no. 5, pp. 201–208, 2021, doi: 10.33166/AETiC.2021.05.025.

[4] N. Thapa, Z. Liu, D. B. Kc, B. Gokaraju, and K. Roy, 'Comparison of Machine Learning and Deep Learning Models for Network Intrusion Detection Systems', pp. 1–16, 2020.

[5] P. Waghmode, M. Kanumuri, H. El-ocla, and T. Boyle, 'Intrusion detection system based on machine learning using least square support vector machine', pp. 1–23, 2025.

[6] V. Z. Mohale and I. C. Obagbuwa, 'Evaluating machine learning-based intrusion detection systems with explainable AI : enhancing transparency and interpretability', no. May, pp. 1–23, 2025, doi: 10.3389/fcomp.2025.1520741.

[7] 'Improving Intrusion Detection with Hybrid Deep Learning Models : A Study on CIC-IDS2017 , UNSW-NB15 , and KDD CUP', vol. 10, 2025.

[8] H. İ. Coşar, 'Intrusion Detection on CSE-CIC-IDS2018 Dataset Using Machine Learning Methods', 2024.

[9] B. Mahbooba, M. Timilsina, R. Sahal, and M. Serrano, 'Explainable Artificial Intelligence (XAI) to Enhance Trust Management in Intrusion Detection Systems Using Decision Tree Model', vol. 2021, 2021, doi: 10.1155/2021/6634811.

[10] D. Gaspar, P. Silva, and C. Silva, 'Explainable AI for Intrusion Detection Systems : LIME and SHAP Applicability on Multi-Layer Perceptron', IEEE Access, vol. 12, no. January, pp. 30164–30175, 2024, doi: 10.1109/ACCESS.2024.3368377.