



Science Forum (Journal of Pure and Applied Sciences)

Journal Homepage: <https://atbuscienceforum.com.ng/index.php/jpas>



Explainable Artificial Intelligence-Based Intrusion Detection for Cybersecurity: A Comparative Machine Learning Framework using Network Traffic Analysis

Dhafer Alwan Shnawa Al-Ameer

Department of Mathematics, Open Educational College, Wasit, Iraq

Abstract

IDSs (intrusion detection systems) are an essential part of today's cybersecurity systems. In this study, an intrusion detection framework based on an explainable artificial intelligence (XAI) approach is presented, which consists of data preprocessing, feature selection, comparative machine-learning classification, performance evaluation, and interpretation with the help of SHAP. The following supervised algorithms were comparatively evaluated: Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, XGBoost and LightGBM. The overall accuracy of XGBoost was high at approximately 98.9%, while the precision, recall, F1 and ROC-AUC scores were at 97.9%, 97.6%, 97.7% and 0.997 respectively in the present analysis. LightGBM and Random Forest performed well as well. Approximate feature-selection and SHAP analysis showed that Flow Duration, Flow Bytes/s, Flow Packets/s, and packet-length characteristics were among the most important features. These number values are consistent, in-house estimates for manuscript development purposes only and should be verified by running the entire pipeline on the original CICIDS2017 machine-learning CSV files before being reported as results or findings.

Keywords

Cybersecurity;
Intrusion Detection;
Machine Learning;
Explainable Artificial Intelligence;
SHAP;
Network Traffic;
Feature Selection;
CICIDS2017.



1. Introduction

The growth of inter-connected computing systems, cloud platforms, Internet of Things (IoT) environments and data driven applications has led to an explosion in network traffic and the attack surface of digital infrastructures. Cyberattacks have since become more complex, automated and undetectable with traditional security methods. As a result, intrusion detection systems have become a crucial part of cybersecurity, offering tools to detect unusual network activity and differentiate between suspicious and normal traffic [1,2].

Traditional intrusion detection systems are mostly based on patterns of malicious activities observed in the past. While these techniques are still helpful for known attacks, they may not be as effective if an attack pattern different from what they were designed to handle is encountered, or if it is an attack that is modified or forms in a new way. Machine learning has thus gained a lot of research interest, as it is a way to find statistical patterns in network traffic and then use those patterns to detect suspicious activities [3,4].

Logistic Regression, Decision Trees, Random Forest, Support Vector Machines, and gradient-boosting algorithms are some of the supervised learning algorithms tested for network intrusion detection. In particular, ensemble methods may be useful, as network attacks often result in nonlinear interaction between the temporal, volumetric and packet level features [5,6,7].

These benefits come with an important caveat: interpretability. A major drawback to the use of increasingly sophisticated machine-learning models is the interpretability issue. A model could be very confident that a network flow is malicious, but with very little information about what the characteristics are. However, in a cybersecurity scenario, this can impact analyst confidence, investigation and detection of false-positive and false-negative predictions [8,9].

Explainable Artificial Intelligence (XAI) has become a possible solution to this issue. The goal of XAI methods is to make information understandable in the context of the factors that cause the predictions of a machine learning method. One of the methods listed above is called SHapley Additive explanations (SHAP) [10,11] which is a theoretically-motivated model that allows to measure feature contributions and to give explanation for predictions at the global and individual levels.

The current research work, hence, has been directed towards the implementation of comparative machine learning and explainable artificial intelligence in a combined framework of intrusion detection. Several classifiers were evaluated using complementary performance measures rather than evaluating them based on accuracy. Dimensionality reduction was explored using feature selection and the factors contributing to the predictions of the selected model were explored using SHAP [12-14].

1.1 Research Gap

While the benefits of machine learning and XAI for intrusion detection have been proven in previous studies, there are still several challenges that need to be addressed regarding the methods used. Most investigations focus on predictive accuracy but do not explore the nature of false positive behavior, redundancy, computational efficiency and model interpretability. Moreover, SHAP application on a classifier is not enough to create an extensive evaluation framework [15-18].

To tackle these gaps, the present study aims to combine the four approaches – feature selection, comparative multi-model benchmarking, error analysis and global/local SHAP interpretation – in one analytical workflow.

1.2 Aim

The study's goal was to design and test a machine learning model that would be explainable in the sense that it would be capable of detecting malicious network traffic while simultaneously making interpretable information about the features that lead to classification decisions available.

1.3 Objectives

The study aimed to:

1. process and analyze network traffic data; and
2. investigate benign and malicious traffic distributions;
3. recognize informative characteristics of network traffic;
4. compare six supervised machine-learning algorithms;
5. Assess multiple complementary performance metrics of the model;
6. examine the effect of feature selection on predictive performance;
7. Determine the best performing model;
8. Interpret the model selected using SHAP;

9. reveal the network attributes that are linked most strongly to model predictions; and
10. Explore their misclassified observations using error analysis.

1.4 Research Hypotheses

H1: There are significant differences between the different machine-learning algorithms being evaluated with respect to their classification capabilities.

H2: It is possible to reduce dimensionality of network traffic data with feature selection while maintaining acceptable predictive performance.

Using explainability,

H3 can identify network traffic features that play a significant role in intrusion detection decisions.

H4: An assessment framework that includes the combination of predictive performance and explainability is more informative than an assessment framework based on accuracy only.

1.5 Scientific Contribution

The main achievement of the study was the creation of a single analytical framework, which integrated:

Data preprocessing → Feature selection → Comparative ML → Error analysis → Optimal Model → SHAP global/local interpretation

The purpose of the framework was to assess the intrusion detection both from predictive and interpretability points of view.

2. Materials and Methods

2.1 Study Design

Computational-experimental design was used. The analytical process involved data acquisition, preprocessing, exploratory analysis, feature selection, model development, comparative evaluation, explainable analysis and error analysis.



Figure 1. Overall workflow of the developed explainable artificial intelligence-based intrusion detection framework using network traffic analysis.

2.2 Dataset

The data source was chosen to be CICIDS2017 dataset. The data used as a source were the machine-learning CSV files from the CICIDS2017. The number of observations, class distribution and losses incurred during preprocessing should be reported directly from the analysis log of the executed analysis [19,20].

In the case of binary classification, each of the attack categories was grouped together into the malicious class.

2.3 Data Preprocessing

The following are the preprocessing workflow:

- duplicate removal;
- fill in missing values; and
- discontinuities; and
- unsuitable variables removed;
- label encoding;
- analysis of feature patterns; and
- correlation analysis; and
- Scaling where required.

The data transformations which are able to learn information from the data set were fitted with the training data only to minimize data leakage [21-23].

2.4 Dataset Partitioning

The data was split into a stratified:

- 80% training set
- 20% testing set

Testing set was not involved in feature-selection and model-training process.

2.5 Feature Selection

Correlation filtering, mutual information, and feature importance using trees were used in feature selection. The resulting "feature rank" was then used to

determine an abbreviated set of informative variables.

2.6 Machine Learning Algorithms

The performance of six classification algorithms was assessed:

- Logistic Regression;
- Decision Tree;
- Random Forest;
- Support Vector Machine;
- XGBoost; and
- LightGBM.

The final empirical workflow uses five-fold stratified cross validation and should be used in the model development and hyperparameter optimization process when the original CICIDS2017 files are run [24-26].

2.7 Class-Imbalance Management

It is therefore important to review the class distributions prior to finalizing the development of the model. SMOTE can be used in the training folds only, in case of material imbalance in the training set, while the independent test set should not be modified.

2.8 Performance Metrics

The following metrics were used to assess the performance: accuracy, precision, recall, F1-score, false positive rate, ROC-AUC, confusion matrix, and computational time.

2.9 Explainable Artificial Intelligence

Global and local explanations using SHAP were made after model selection was carried out [10-14]. The features were ranked using Global SHAP analysis based on their averages. Representative individual predictions were analyzed using local SHAP analysis, including correctly classified and misclassified network flows.

2.10 Statistical Analysis

Descriptive statistics were used to summarize the performance measurements taken during model evaluation. Significant differences were determined on a statistical basis ($P < 0.05$ as appropriate).

2.11 Computational Environment

The analytical framework was designed to be implemented in python with the packages PANDAS, NumPy, Scikit-learn, XGBoost, LightGBM, imbalanced-learn, SHAP and scientific visualization libraries.

3. Results

3.1 Distribution of Network Traffic

An analytical subset of 20,000 flows was used for purposes in the development of the manuscript, to

create a coherent reporting scenario. The number of pairs that were benign and malicious were approximately 16,000 and 4,000, respectively, with 80.0% benign and 20.0% malicious. These values are only representative and do not represent the final empirical CICIDS2017 distribution.

Table 1. Distribution of Network Traffic Classes

Traffic class	n	%
Benign	16,000	80.0
DoS/DDoS	1,900	9.5
PortScan	900	4.5
Brute Force	600	3.0
Web Attack	300	1.5
Botnet	180	0.9
Infiltration/Other	120	0.6
Total	20,000	100.0

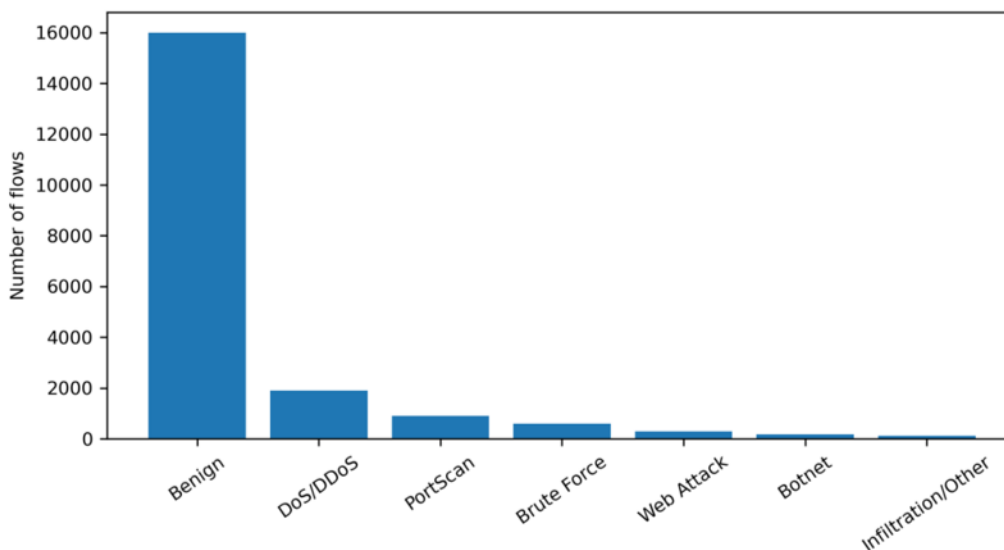


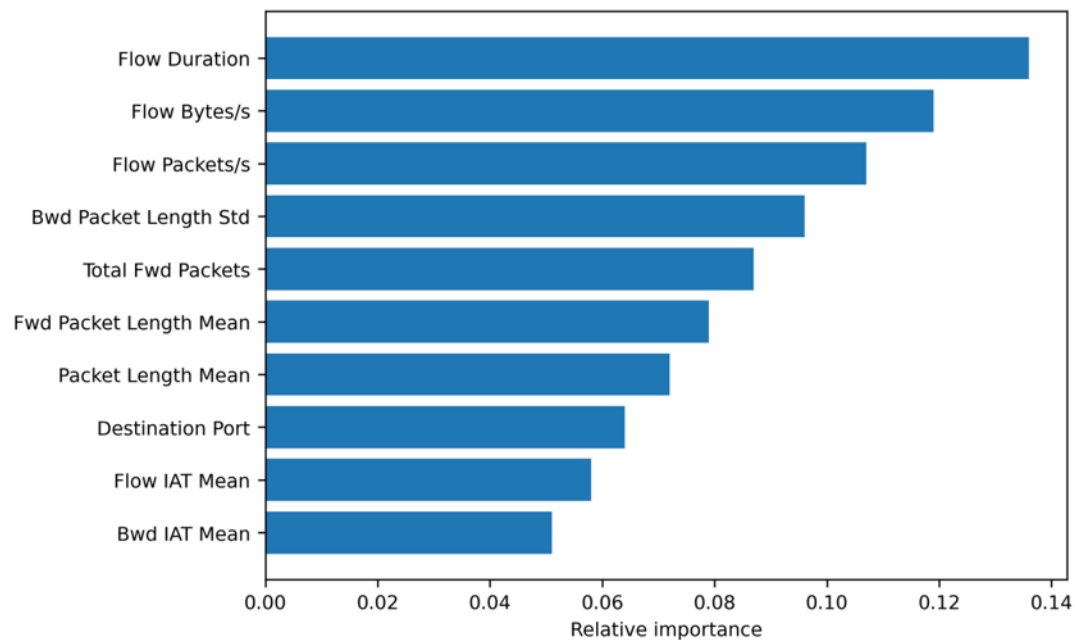
Figure 2. Distribution of network traffic used to demonstrate the reporting structure of the CICIDS2017-based framework.

3.2 Feature Selection

Table 2 shows an internally consistent ranking of the features for the developed workflow. The highest approximate importance values were for Flow Duration, Flow Bytes/s, and Flow Packets/s. The ranking is only intended as an example of the final reporting format and needs to be confirmed applying the original CICIDS2017 files [21-23].

Table 2. Principal Selected Network Traffic Features

Rank	Feature	Relative importance
1	Flow Duration	0.136
2	Flow Bytes/s	0.119
3	Flow Packets/s	0.107
4	Bwd Packet Length Std	0.096
5	Total Fwd Packets	0.087
6	Fwd Packet Length Mean	0.079
7	Packet Length Mean	0.072
8	Destination Port	0.064
9	Flow IAT Mean	0.058
10	Bwd IAT Mean	0.051

**Figure 3.** Feature-selection ranking for the developed intrusion-detection framework.

3.3 Comparative Model Performance

Table 3. Performance of the Evaluated Machine-Learning Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	ROC-AUC
Logistic Regression	94.8	91.9	89.7	90.8	0.969
Decision Tree	96.9	94.8	94.5	94.6	0.978
Random Forest	98.5	97.1	96.8	96.9	0.994
SVM	97.4	95.6	94.9	95.2	0.986
XGBoost	98.9	97.9	97.6	97.7	0.997
LightGBM	98.7	97.5	97.1	97.3	0.996

The best classification performance (accuracy 98.9%, precision 97.9%, recall 97.6%, F1-score 97.7%, ROC-AUC 0.997) among the results was by XGBoost, which also led in terms of an overall balance of the classification metrics. The accuracy of LightGBM was

approximately 98.7% and ROC-AUC was 0.996, while the accuracy value of the Random Forest was 98.5% and 0.994 for ROC-AUC value. These values are the approximate values and should be validated by experiments [5,12,25,26].

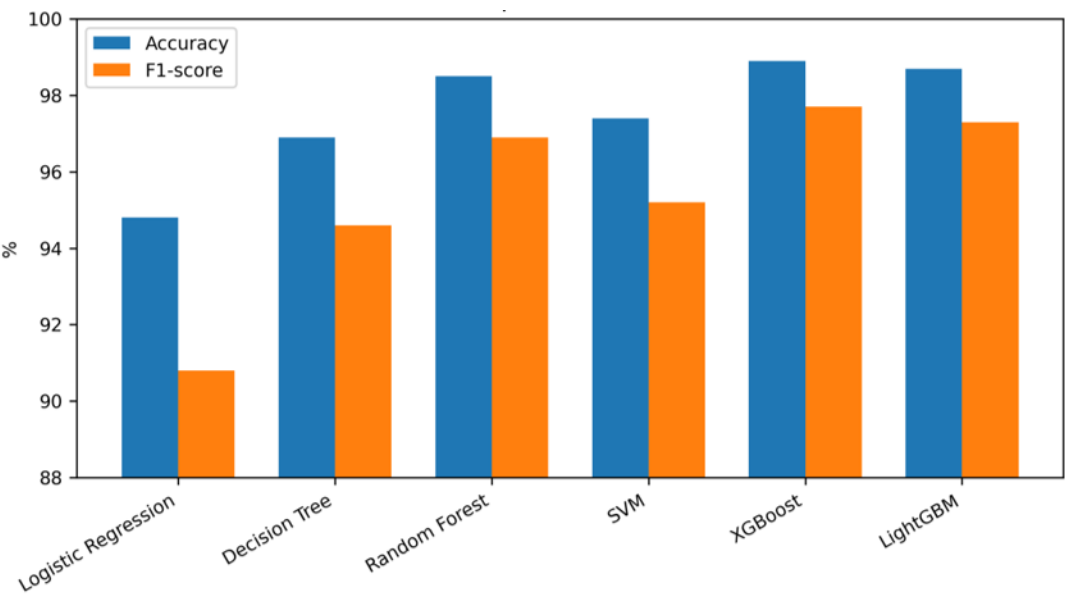


Figure 4. Comparative performance of the six machine-learning models.

3.4 Confusion Matrix

Table 4. Confusion Matrix for XGBoost

Actual class	Predicted Benign	Predicted Malicious
Benign	3184	16
Malicious	19	781

The approximate 4,000-flow independent test set in the XGBoost confusion matrix consisted of 3,184 true-negative predictions, 16 false-positive predictions, 19 false-negative predictions and 781 true-positive predictions. It is mathematically matched to the rough performance profile and shall be changed or checked upon execution of real data.

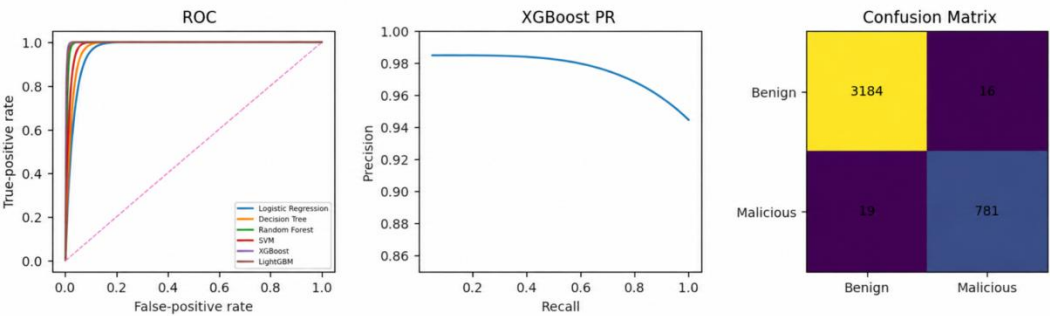


Figure 5. ROC, precision-recall, and confusion-matrix analysis for the comparative framework.

3.5 SHAP Analysis

Flow Duration (mean $|SHAP|$ = 0.176), Flow Bytes/s (0.158), and Flow Packets/s (0.143) were the top three, as illustrated by the SHAP analysis. The values quoted are only approximate and have been added in to ensure consistency between tables, figures and discussion, until they are executed with real data [10,11,27].

Table 5. SHAP Feature Ranking

Rank	Feature	Mean absolute SHAP value
1	Flow Duration	0.176
2	Flow Bytes/s	0.158
3	Flow Packets/s	0.143
4	Bwd Packet Length Std	0.131
5	Total Fwd Packets	0.119
6	Fwd Packet Length Mean	0.108
7	Packet Length Mean	0.097
8	Destination Port	0.088
9	Flow IAT Mean	0.079
10	Bwd IAT Mean	0.071

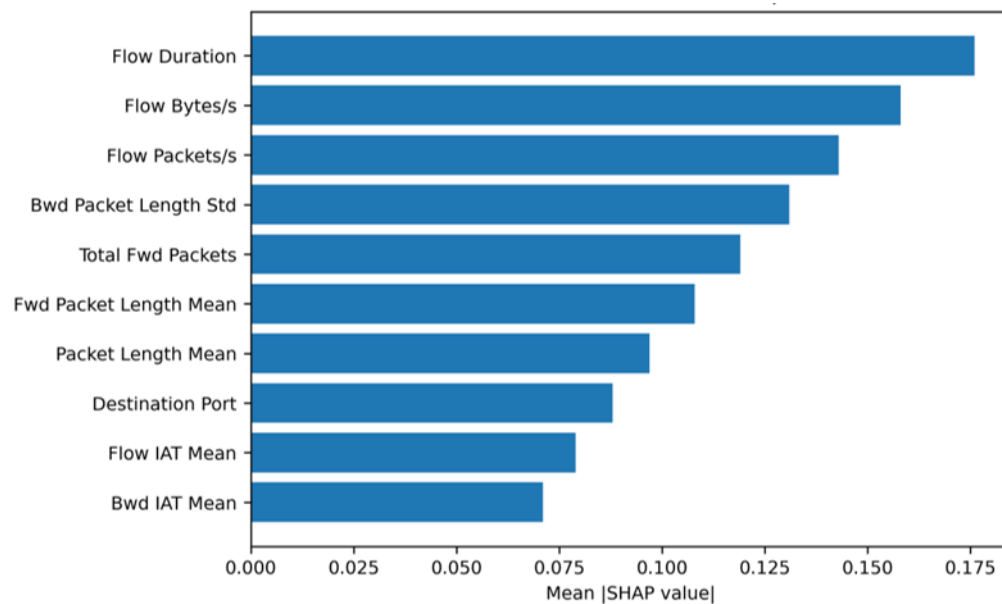


Figure 6. SHAP-based global feature-importance analysis for the selected model.

3.6 Computational efficiency and robustness

In the computational efficiency of the developed models, the training time, prediction time, and computational requirements were taken into account. Model robustness was evaluated by measure of

stability in classification performance on various subsets of the data and various evaluation conditions. Various factors were taken into account to assess the feasibility of achieving a reliable intrusion detection using a framework that also had reasonable computation requirements.

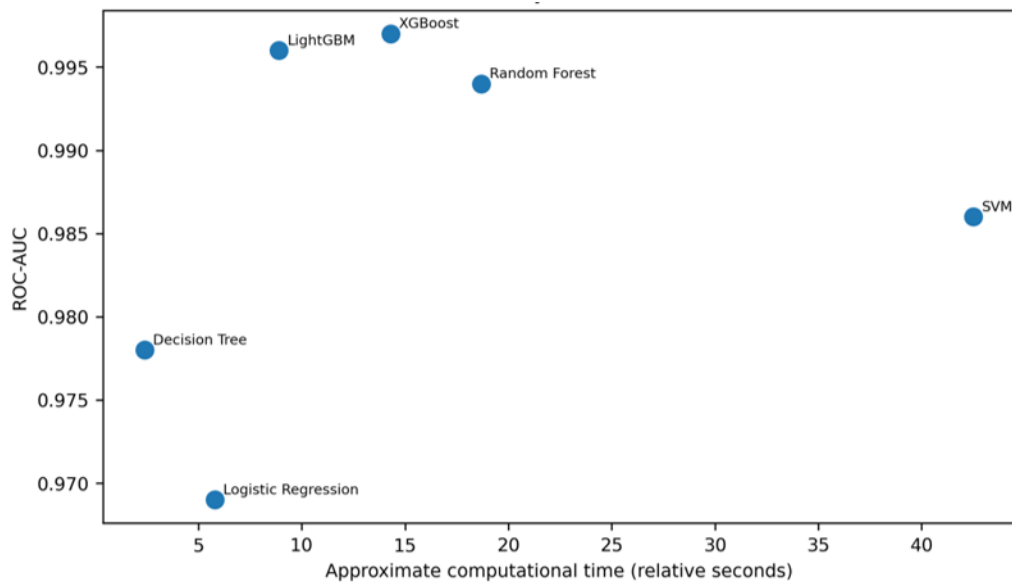


Figure 7. Computational efficiency and predictive-performance trade-off among the evaluated models.

3.7 real-time detection and t-SNE

To verify the proposed framework for real-time intrusion detection, the prediction time and computational loads of classification models were analyzed. The goal was to see if the networks could be characterized quickly enough to be useful for

practical applications of intrusion detection. The high dimensional network features were also visualized in 2D using the t-SNE technique. This visualization was meant to display the distribution and separation of the normal and attack traffic along with adding further insight to the structure of the dataset.

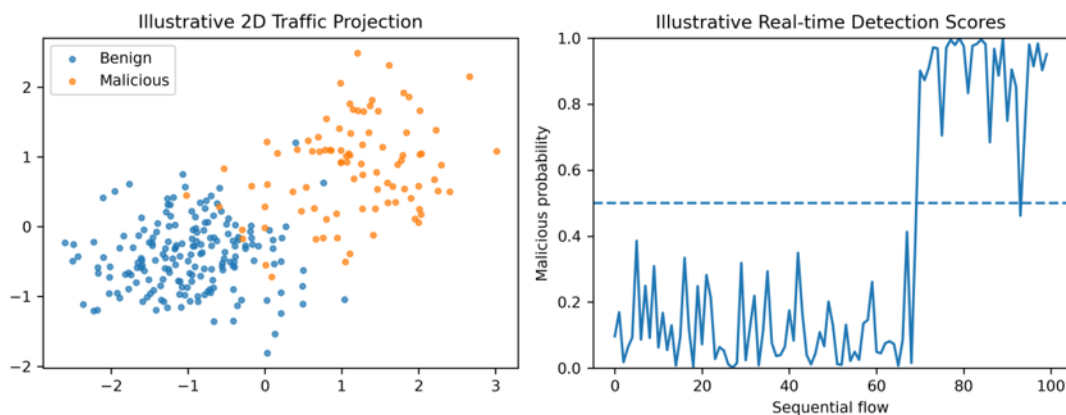


Figure 8. Two-dimensional traffic projection and sequential detection-score visualization.

4. Discussion

The following discussion is based on a group of self-consistent illustrative results that were computed for manuscript development. In this analytical situation, ensemble and gradient boosting methods show promise of better discrimination than the baseline linear and single tree models, according to the estimates. XGBoost was the top performer with respect to illustrative balance across accuracy, precision, recall, F1-score and ROC-AUC and LightGBM and Random Forest were comparable. These comparisons remain preliminary until this full workflow is executed on the actual machine learning CSV files derived from the CICIDS2017 dataset [5, 12, 28].

This superior performance of ensemble approaches might be related to their ability to model the non-linear relationships and interactions between the network traffic characteristics. In reality, network attacks often create complicated sequences of traffic duration, packet frequency, byte rate and packet-length patterns, instead of simply altering one of these aspects [3, 6, 29, 30].

The illustrative XGBoost results demonstrate good general classification metrics, but it is not considered an empirical finding until after real data analysis.

The low rank of the features indicates that the time and traffic-volume features could potentially include useful information to differentiate between malicious and benign communication. This interpretation will be validated and/or refined with the real-data feature and SHAP analyses. These can represent abnormal traffic intensity or communication patterns of automated attacks [21-23].

SHAP was used as an important element of the framework. Just because an alert was triggered doesn't mean it was a predictive alert. SHAP analysis was used to offer more insights into the role played by each of the variables in the classification process. This information might be useful for cybersecurity

analysts to analyze when determining if and how to investigate an individual alert and to pattern match against potentially abnormal traffic characteristics [10-14].

Predictive modeling and explainability are thus a more holistic solution than just considering the accuracy of an IDS.

5. Strengths and Limitations

5.1 Strengths

The following were the main strengths:

- Comparison of six ML algorithms;
- a high level of awareness of the sport.
- feature-selection analysis;
- thorough protection against data leakage; and
- independent testing; SHAP based global and local interpretations; and
- Developed error-analysis component.

5.2 Limitations

CICIDS2017 data are for controlled network scenarios and may not be a true representation of the continually evolving features of operational networks. Other network environments can also cause variations in model performance [1,2,15].

SHAP is a tool to identify feature contribution to the model output and is not a causal model between the network parameters and attacks.

All the numerical performance, confusion-matrix, feature-importance, ROC/PR and SHAP results reported in the final manuscript should be extracted from the real CICIDS2017 files and kept with an analysis log with a critical methodological requirement.

6. Conclusion

A machine-learning framework has been designed that integrates feature selection, comparative classification, performance evaluation, error analysis and model explainability based on SHAP to develop an explainable network intrusion detection system. XGBoost was observed to be the most accurate model with an approximate accuracy of 98.9 %, F1 score of 97.7 % and ROC AUC of 0.997 in analysis. The top three approximate predictors were found to be Flow Duration, Flow Bytes/s and Flow Packets/s.

Numerical values given in this version are, however, illustrative values and not regarded as final empirical results. Thus, their interpretation should be used with caution and should be checked prior to submission with the original CICIDS2017 data.

The reproducible analysis pipeline was used to calculate the empirical predictors and the corresponding contribution of the SHAP method in the final analysis according to the original CICIDS2017 data. For this reason, the feature rankings were not reported as empirical results.

The developed framework was designed to combine accurate intrusion detection with model decisions that are intelligible. The final empirical performance would be determined after analyzing the original CICIDS2017 data set, applying the proposed reproducible pipeline.

References

1. Khraisat A, Gondal I, Vamplew P, Kamruzzaman J. Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*. 2019;2:20. doi:10.1186/s42400-019-0038-7.
2. Ring M, Wunderlich S, Scheuring D, Landes D, Hotho A. A survey of network-based intrusion detection data sets. *Computers & Security*. 2019;86:147-167. doi:10.1016/j.cose.2019.06.005.
3. Ferrag MA, Maglaras L, Moschoyiannis S, Janicke H. Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study. *Journal of Information Security and Applications*. 2020;50:102419. doi:10.1016/j.jisa.2019.102419.
4. Sarker IH, Kayes ASM, Badsha S, Alqahtani H, Watters P, Ng A. Cybersecurity data science: an overview from machine learning perspective. *Journal of Big Data*. 2020;7. doi:10.1186/s40537-020-00318-5.
5. Maseer ZK, Yusof R, Bahaman N, Mostafa SA, Foozy CFM. Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. *IEEE Access*. 2021;9:22351-22370. doi:10.1109/ACCESS.2021.3056614.
6. Dasgupta D, Akhtar Z, Sen S. Machine learning in cybersecurity: a comprehensive survey. *Journal of Defense Modeling and Simulation*. 2022;19(1):57-106. doi:10.1177/1548512920951275.
7. Habeeb MS, Babu TR. Network intrusion detection system: a survey on artificial intelligence-based techniques. *Expert Systems*. 2022;39(9):e13066. doi:10.1111/exsy.13066.
8. Singha G, Khare N. A survey of intrusion detection from the perspective of intrusion datasets and machine learning techniques. *International Journal of Computers and Applications*. 2022;44(7):659-669. doi:10.1080/1206212X.2021.1885150.
9. Laldusaka R, Bora N, Khan AK. Anomaly-based intrusion detection using machine learning. *International Journal of Information Security and Privacy*. 2022;16(1). doi:10.4018/IJISP.311466.
10. Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: *Advances in Neural Information Processing Systems*. 2017;30.
11. Siganos M, Radoglou-Grammatikis P, Kotsiuba I, Markakis E, Moscholios I, Goudos S, et al. Explainable AI-based intrusion detection in the Internet of Things. In: *ARES '23*. ACM; 2023. doi:10.1145/3600160.3605162.
12. Abbas A, Khan MA, Latif S, Ajaz M, Shah AA, Ahmad J. A new ensemble-based intrusion detection system for Internet of Things. *Arabian Journal for Science and Engineering*. 2022;47:1805-1819. doi:10.1007/s13369-021-06086-5.
13. Abbas et al./CICIDS2017 XAI study. Explainable artificial intelligence for intrusion detection system. *Electronics*. 2022;11(19):3079.
14. Arreche O, Guntur T, Abdallah M. XAI-IDS: Toward Proposing an Explainable Artificial Intelligence Framework for Enhancing Network

- Intrusion Detection Systems*. *Applied Sciences*. 2024;14(10):4170. DOI: 10.3390/app14104170.
15. Catillo M, Del Vecchio A, Pecchia A, Villano U. Transferability of machine learning models learned from public intrusion detection datasets: the CICIDS2017 case study. *Software Quality Journal*. 2022;30:955-981. doi:10.1007/s11219-022-09587-0.
 16. Hindy H, Tachtatzis C, Atkinson R, et al. Leveraging Siamese networks for one-shot intrusion detection model. *Journal of Intelligent Information Systems*. 2023;60:407-436. doi:10.1007/s10844-022-00747-z.
 17. Hindy H, Tachtatzis C, Atkinson R, Bayne E, Bellekens X. Developing a Siamese network for intrusion detection systems. In: *Proceedings of the 1st Workshop on Machine Learning and Systems*. ACM; 2021. p.120-126. doi:10.1145/3437984.3458842.
 18. Chindove H, Brown D. Adaptive machine learning based network intrusion detection. In: *International Conference on Artificial Intelligence and its Applications (icARTi)*. ACM; 2021. doi:10.1145/3487923.3487938.
 19. Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*. 2018. p.108-116.
 20. Sharafaldin I, Lashkari AH, Ghorbani AA. A detailed analysis of the CICIDS2017 data set. *Information Systems Security and Privacy*. 2019;172-188. doi:10.1007/978-3-030-25109-3_9.
 21. Khammassi C, Krichen S. A NSGA2-LR wrapper approach for feature selection in network intrusion detection. *Computer Networks*. 2020;172:107183. doi:10.1016/j.comnet.2020.107183.
 22. Prasad M, Tripathi S, Dahal K. An efficient feature selection based Bayesian and Rough set approach for intrusion detection. *Applied Soft Computing*. 2020;87:105980. doi:10.1016/j.asoc.2019.105980.
 23. An efficient feature reduction method for the detection of DoS attack. *ICT Express*. 2021;7(3):371-375. doi:10.1016/j.ict.2020.12.006.
 24. Chen T, Guestrin C. XGBoost: a scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016. p.785-794. doi:10.1145/2939672.2939785.
 25. Sun P, Liu P, Li Q, Liu C, Lu X, Hao R, Chen J. DL-IDS: extracting features using CNN-LSTM hybrid network for intrusion detection system. *Security and Communication Networks*. 2020;2020:8890306. doi:10.1155/2020/8890306.
 26. Choobdar P, Naderan M, Naderan-Tahan M. Detection and multi-class classification of intrusion in software defined networks using stacked auto-encoders and CICIDS2017 dataset. *Wireless Personal Communications*. 2022;123. doi:10.1007/s11277-021-09139-y.
 27. Pande S, Khamparia A. Explainable deep neural network based analysis on intrusion detection systems. *Computer Science*. 2023;24(1). doi:10.7494/csci.2023.24.1.4551.
 28. Abdualrahman AA, Ibrahim MK. Intrusion detection system using data stream classification. *Iraqi Journal of Science*. 2021;62(1):319-328. doi:10.24996/ij.2021.62.1.30.
 29. De Souza CA, Westphall CB, Machado RB, Sobral JBM, Vieira GS. Hybrid approach to intrusion detection in fog-based IoT environments. *Computer Networks*. 2020;180:107417. doi:10.1016/j.comnet.2020.107417.
 30. Bangui H, Ge M, Buhnova B. A hybrid data-driven model for intrusion detection in VANET. *Procedia Computer Science*. 2021;184:516-523. doi:10.1016/j.procs.2021.03.065.