

A hybrid explainable machine learning and deep learning framework for IoT network anomaly detection

Mussa Turdalyuly^{1,†}, Nazerke Baisholan^{2*,†}, Sergiy Gnatyuk^{3,†} and Karlygash Baisholanova^{4,†}

¹META University, Dosmukhamedov Str., 45, Almaty, 050012, Kazakhstan

²International Engineering and Technological University, Al-Farabi Ave., 89/21, Almaty, 050060, Kazakhstan

³National University "Kyiv Aviation Institute", Liubomyra Huzara Ave., 1, Kyiv, 03058, Ukraine

⁴Al-Farabi Kazakh National University, Al-Farabi Ave., 71, Almaty, 050040, Kazakhstan

Abstract

The rapid growth of Internet of Things (IoT) infrastructures has increased the need for accurate and interpretable network anomaly detection systems. This study presents a hybrid explainable machine learning and deep learning framework for detecting malicious IoT network traffic and classifying attack categories. The CICIoT2023 dataset was used as an experimental benchmark. The proposed pipeline includes binary anomaly detection, multiclass attack classification, comparison of classical machine learning and deep learning models, and explainability analysis using Random Forest feature importance, permutation importance, and SHAP values. Four machine learning models, namely Random Forest, Extra Trees, XGBoost, and LightGBM, were compared with two deep learning models, MLP and 1D-CNN. Experimental results showed that Random Forest achieved the best multiclass classification performance with an accuracy of 0.9961, macro-F1 score of 0.8698, and weighted-F1 score of 0.9960. The binary anomaly detection task also demonstrated strong performance, with an attack F1-score of 0.9988 and ROC-AUC of 0.9998. Deep learning models trained on GPU achieved competitive overall accuracy but did not outperform tree-based ensemble models on tabular network-flow features. Explainability analysis revealed that inter-arrival time was the most influential feature across all interpretation methods, indicating that temporal packet-arrival behavior plays a dominant role in IoT attack classification. The results demonstrate that explainable ensemble learning can provide both high detection performance and interpretable insights for IoT cybersecurity applications.

Keywords

IoT security, network anomaly detection, intrusion detection, machine learning, deep learning, Random Forest, explainable AI, SHAP, CICIoT2023

1. Introduction

The rapid expansion of Internet of Things (IoT) systems has significantly increased the complexity and vulnerability of modern network infrastructures. IoT devices are widely deployed in smart homes, healthcare systems, industrial environments, transportation, and critical infrastructure. However, many IoT devices have limited computational resources, weak security mechanisms, and heterogeneous communication protocols, making them attractive targets for cyberattacks.

Network anomaly detection is one of the key components of IoT cybersecurity. Traditional rule-based intrusion detection systems often fail to detect new or evolving attack patterns because they rely on predefined signatures. In contrast, machine learning and deep learning methods can learn complex behavioral patterns from network traffic data and distinguish between benign and malicious activities. Nevertheless, the practical use of such models in cybersecurity requires not only high accuracy but also interpretability. Security analysts need to understand which traffic features contribute to model decisions in order to trust and validate automated detection systems.

CMiGIN'26: Fifth International Workshop on Conflict Management in Global Information Networks, June 26–27, 2026, Krakow, Poland

*Corresponding author.

†These authors contributed equally.

✉ m.turdalyuly@gmail.com (M. Turdalyuly); nbaishol@purdue.edu (N. Baisholan); s.gnatyuk@kai.edu.ua (S. Gnatyuk); baisholanova.k@gmail.com (K. Baisholanova)

ORCID 0000-0002-1470-3706 (M. Turdalyuly); 0000-0002-8134-0466 (N. Baisholan); 0000-0002-9421-8566 (S. Gnatyuk); 0000-0001-7375-5998 (K. Baisholanova)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

This study proposes a hybrid explainable machine learning and deep learning framework for IoT network anomaly detection. The framework is evaluated on the CICIoT2023 dataset, which was designed as a large-scale benchmark for IoT attack detection and includes 33 attacks executed in an IoT topology composed of 105 devices [1, 2]. The dataset groups attacks into seven categories: DDoS, DoS, Recon, Web-based, Brute Force, Spoofing, and Mirai [1, 2]. In this work, two tasks are considered: binary classification of benign and malicious traffic, and multiclass classification of attack categories.

The main contributions of this study are as follows.

- A unified experimental framework for binary anomaly detection and multiclass attack classification in IoT traffic is developed.
- Classical machine learning models and GPU-trained deep learning models are compared under the same preprocessing and evaluation conditions.
- The effect of class imbalance on multiclass attack classification is analyzed using macro-F1, weighted-F1, and per-class metrics.
- Explainability analysis is performed using Random Forest feature importance, permutation importance, and SHAP values.
- The study demonstrates that tree-based ensemble models outperform neural networks on tabular CICIoT2023 flow features.

2. Related work

Machine learning-based intrusion detection has been widely studied for traditional, IoT, and Industrial IoT networks. Earlier benchmark datasets such as CICIDS2017, UNSW-NB15, Bot-IoT, and ToN-IoT played an important role in evaluating intrusion detection systems under different network conditions [3, 4, 5, 6]. However, many earlier datasets either do not focus specifically on IoT devices or do not fully represent large-scale IoT environments with diverse real-device topologies.

The CICIoT2023 dataset was introduced to address this limitation by providing a large-scale and realistic IoT attack environment. According to its original description, 33 attacks were executed in an IoT topology composed of 105 devices, and the attacks were grouped into seven categories: DDoS, DoS, Recon, Web-based, Brute Force, Spoofing, and Mirai [1, 2]. This makes CICIoT2023 suitable for both binary anomaly detection and multiclass attack classification in IoT traffic.

Classical machine learning models, especially tree-based ensemble methods, remain highly effective for tabular network-flow features. Random Forest [7], Extra Trees [8], XGBoost [9], and LightGBM [10] are widely used because of their strong predictive performance, scalability, and ability to model non-linear feature interactions. At the same time, deep learning models, including multilayer perceptrons, convolutional neural networks, recurrent neural networks, and autoencoders, have been actively applied to intrusion detection tasks [11, 12, 13, 14]. Recent studies have continued to explore deep learning-based IoT intrusion detection, including CNN-based and hybrid architectures for improving detection accuracy and generalization [15, 16].

Recent research has also emphasized the need for explainable intrusion detection systems. Gaspar et al. [17] investigated the applicability of LIME and SHAP for explaining multilayer perceptron-based intrusion detection models. Alabbadi and Bajaber [18] proposed an explainable intrusion detection system over IoT data streams using XAI techniques. Ebrahimi et al. [15] developed a CNN-based IoT intrusion detection approach with explainability analysis, while Mohale and Obagbuwa [19] evaluated explainable machine learning-based IDS models to improve transparency and interpretability. These recent works confirm that high detection accuracy alone is insufficient for practical cybersecurity applications, where analysts need interpretable evidence supporting model decisions.

Another major challenge in intrusion detection is class imbalance. Cybersecurity datasets often contain dominant attack classes and rare minority classes, which can lead to misleadingly high accuracy while reducing detection quality for underrepresented attacks. Classical studies on imbalanced learning and oversampling methods remain relevant for this problem [20, 21]. Recent studies on imbalanced

anomaly detection and interpretable fraud detection have also shown the importance of macro-level metrics, class-aware evaluation, and explainable ensemble learning [22, 23, 24]. Although FraudX AI was proposed in the financial fraud domain, its focus on imbalanced data, interpretability, and ensemble learning is directly relevant to IoT intrusion detection, where minority attack categories may be difficult to identify.

In summary, recent literature shows three important tendencies: the growing use of realistic IoT-specific datasets, the increasing role of explainable AI in IDS, and the importance of evaluating models under class imbalance. Therefore, this study combines machine learning, deep learning, class-imbalance-aware evaluation, and explainability analysis in a unified framework for IoT network anomaly detection.

3. Materials and methods

3.1. Dataset description

The CICIoT2023 dataset was used as the experimental benchmark in this study. It contains IoT network traffic collected from a realistic IoT topology and includes multiple attack scenarios. According to the dataset description, 33 attacks were executed in an IoT topology composed of 105 devices, and attacks were grouped into seven categories: DDoS, DoS, Recon, Web-based, Brute Force, Spoofing, and Mirai [1, 2].

In this study, two classification tasks were considered. The first task was binary anomaly detection, where all attack categories were mapped to a single Attack class and compared against Benign traffic. The second task was multiclass attack classification, where the model predicted one of the following classes: Benign, Brute Force, DDoS, DoS, Mirai, Recon, Spoofing, and Web-based.

3.2. Data preprocessing

The dataset was provided in train, validation, and test CSV files. The original split was preserved to avoid data leakage. Non-numeric columns, except the target label, were removed from the feature matrix. Infinite values were replaced with missing values, and missing values were imputed using the median values computed from the training set. For deep learning models, numerical features were standardized using the StandardScaler fitted only on the training data.

The original attack labels were mapped into broader attack categories. The BrowserHijacking label was assigned to the Web-based category because it represents a browser-level web attack. This mapping removed the ambiguous Other class and resulted in eight final classes including Benign traffic.

3.3. Models

The following classical machine learning models were evaluated: Random Forest, Extra Trees, XGBoost, and LightGBM. These models were selected because tree-based ensemble methods are widely used for structured tabular data and often demonstrate strong performance in network-flow classification tasks [7, 8, 9, 10].

Two deep learning models were also implemented: a multilayer perceptron (MLP-v2) and a one-dimensional convolutional neural network (1D-CNN). The neural models were trained using PyTorch [25]. To address class imbalance, class-weighted cross-entropy loss was used. Instead of applying full balanced weights, softened class weights were computed using the square root of balanced weights and clipping to avoid over-amplification of rare classes.

3.4. Evaluation metrics

The models were evaluated using accuracy, precision, recall, F1-score, macro-F1, weighted-F1, ROC-AUC for binary classification, and training time. For multiclass classification, macro-F1 was considered the primary metric because the dataset is highly imbalanced and minority attack classes should not

be ignored. Weighted-F1 was also reported to show overall performance while accounting for class support.

4. Experimental setup

The experiments were conducted on a local high-performance computing server equipped with three NVIDIA L40 GPUs, each with 40 GB of GPU memory. Classical machine learning models were trained on CPU, while deep learning models were trained using GPU acceleration. The implementation used Python, scikit-learn [26], XGBoost [9], LightGBM [10], PyTorch [25], and SHAP [27]. Permutation importance was used as a model-agnostic feature analysis method following the variable importance framework described by Fisher et al. [28].

5. Results

5.1. Binary anomaly detection

For the binary anomaly detection task, Random Forest achieved the best performance, as shown in Table 1. It obtained an accuracy of 0.9976, attack precision of 0.9989, attack recall of 0.9986, attack F1-score of 0.9988, and ROC-AUC of 0.9998. These results indicate that the model is highly effective in distinguishing malicious IoT network traffic from benign traffic. The corresponding confusion matrix is presented in Figure 1, where the model correctly classified 1,147,578 attack samples and 26,411 benign samples.

Table 1
Binary Anomaly Detection Results.

Model	Accuracy	Attack Precision	Attack Recall	Attack F1	ROC-AUC
RandomForest	0.9976	0.9989	0.9986	0.9988	0.9998
ExtraTrees	0.9973	0.9989	0.9983	0.9986	0.9997
XGBoost	0.9969	0.9990	0.9978	0.9984	0.9996
LightGBM	0.9947	1.0000	0.9946	0.9973	0.9995

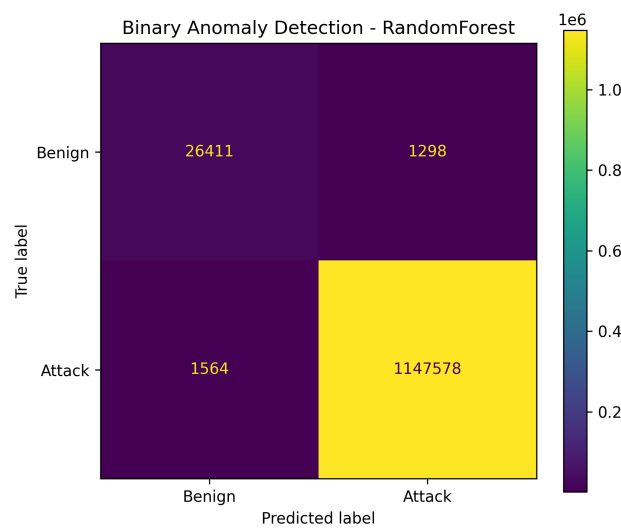


Figure 1: Confusion matrix for binary anomaly detection using the Random Forest model.

5.2. Multiclass attack classification

The multiclass attack classification results are summarized in Table 2. Random Forest achieved the best overall performance, with an accuracy of 0.9961, macro-F1 score of 0.8698, and weighted-F1 score of 0.9960. Extra Trees achieved the second-best macro-F1 score of 0.8276. XGBoost and LightGBM also demonstrated high weighted-F1 scores but lower macro-F1 values, indicating weaker performance on minority classes. Figure 2 visually compares all evaluated models according to macro-F1 score and shows that tree-based ensemble models outperformed the GPU-trained neural models.

Table 2

Multiclass Attack Classification Results.

Model	Accuracy	Macro-F1	Weighted-F1	Training time, s
RandomForest	0.9961	0.8698	0.9960	113.93
ExtraTrees	0.9947	0.8276	0.9945	153.60
XGBoost	0.9951	0.7979	0.9949	179.62
LightGBM	0.9937	0.7780	0.9942	141.13
MLP-v2	0.9912	0.7688	0.9911	787.55
1D-CNN	0.9868	0.6994	0.9866	837.52

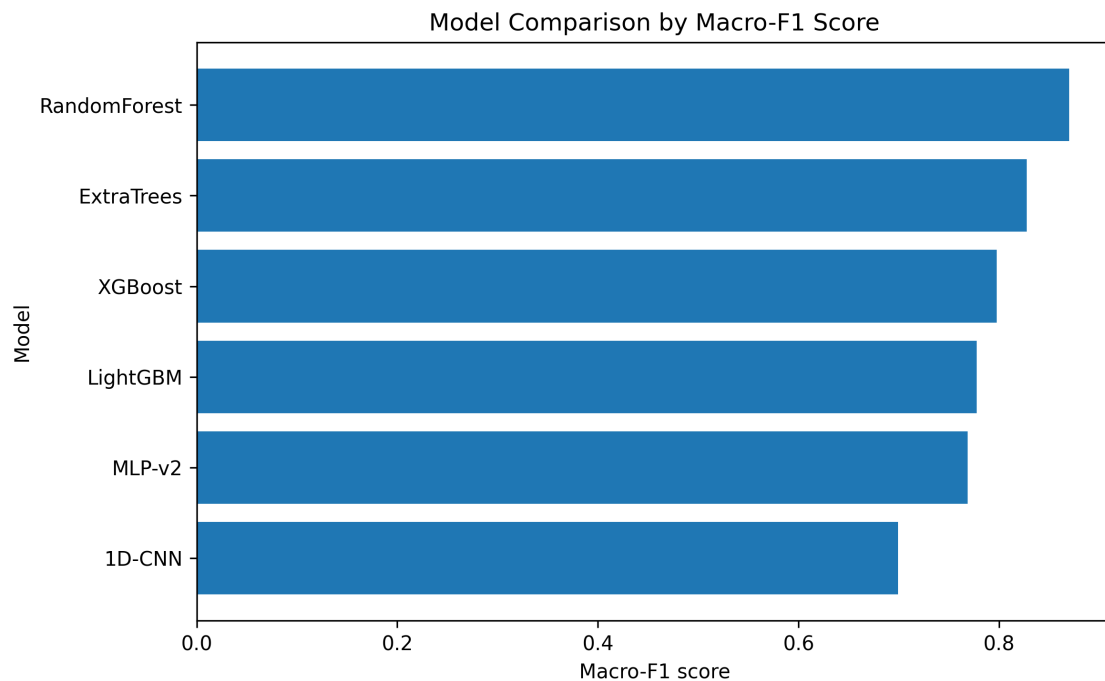


Figure 2: Comparison of machine learning and deep learning models by macro-F1 score.

5.3. Per-class performance

The per-class evaluation of the best Random Forest model is reported in Table 3. The model achieved near-perfect F1-scores for dominant attack categories such as DDoS, DoS, and Mirai. However, minority classes such as Brute Force and Web-based remained more difficult to classify, with F1-scores of 0.7214 and 0.5417, respectively. This confirms that high overall accuracy and weighted-F1 may hide reduced performance on underrepresented attack categories.

Table 3

Per-class Performance of the Best Random Forest Model.

Class	Precision	Recall	F1-score	Support
Benign	0.9110	0.9803	0.9444	27,709
Brute Force	1.0000	0.5643	0.7214	319
DDoS	0.9997	1.0000	0.9998	855,981
DoS	0.9999	0.9994	0.9996	204,245
Mirai	0.9999	0.9990	0.9994	66,749
Recon	0.9096	0.8201	0.8626	8,812
Spoofing	0.9181	0.8623	0.8893	12,410
Web-based	0.9832	0.3738	0.5417	626

6. Explainability analysis

Explainability analysis was conducted for the best-performing Random Forest model. Three interpretation methods were used: impurity-based Random Forest feature importance, permutation importance [28], and SHAP values [27].

The analysis showed that IAT was the dominant feature across all three methods. Random Forest feature importance assigned IAT the highest importance score of 0.2137. Permutation importance showed that permuting IAT caused the macro-F1 score to decrease by 0.2902, which was the largest drop among all features. SHAP analysis also identified IAT as the most influential feature, with a mean absolute SHAP value of 0.0652.

Other important features included rst_count, urg_count, flow_duration, Variance, syn_count, Rate, Srate, Magnitude, and Tot size. These features describe temporal traffic behavior, TCP flag-related patterns, packet-size statistics, and traffic intensity. The consistency between the three explainability methods indicates that the model relies on meaningful network-flow characteristics rather than arbitrary correlations. The numerical results of the three explainability methods are summarized in Table 4. In addition, Figure 3 presents the SHAP-based global feature importance of the Random Forest model.

Table 4

Top Explainability Features of the Random Forest Model.

Feature	RF importance	Permutation importance	Mean SHAP
IAT	0.2137	0.2902	0.0652
rst_count	0.0549	0.0178	0.0167
urg_count	0.0381	0.0220	0.0087
flow_duration	0.0448	0.0101	0.0085
Variance	0.0316	0.0064	0.0182
syn_count	0.0235	0.0062	0.0060
Rate	0.0299	0.0054	0.0029
Duration	0.0213	0.0051	0.0027
Srate	0.0307	0.0039	0.0031
Magnitue (Magnitude)	0.0511	0.0035	0.0225

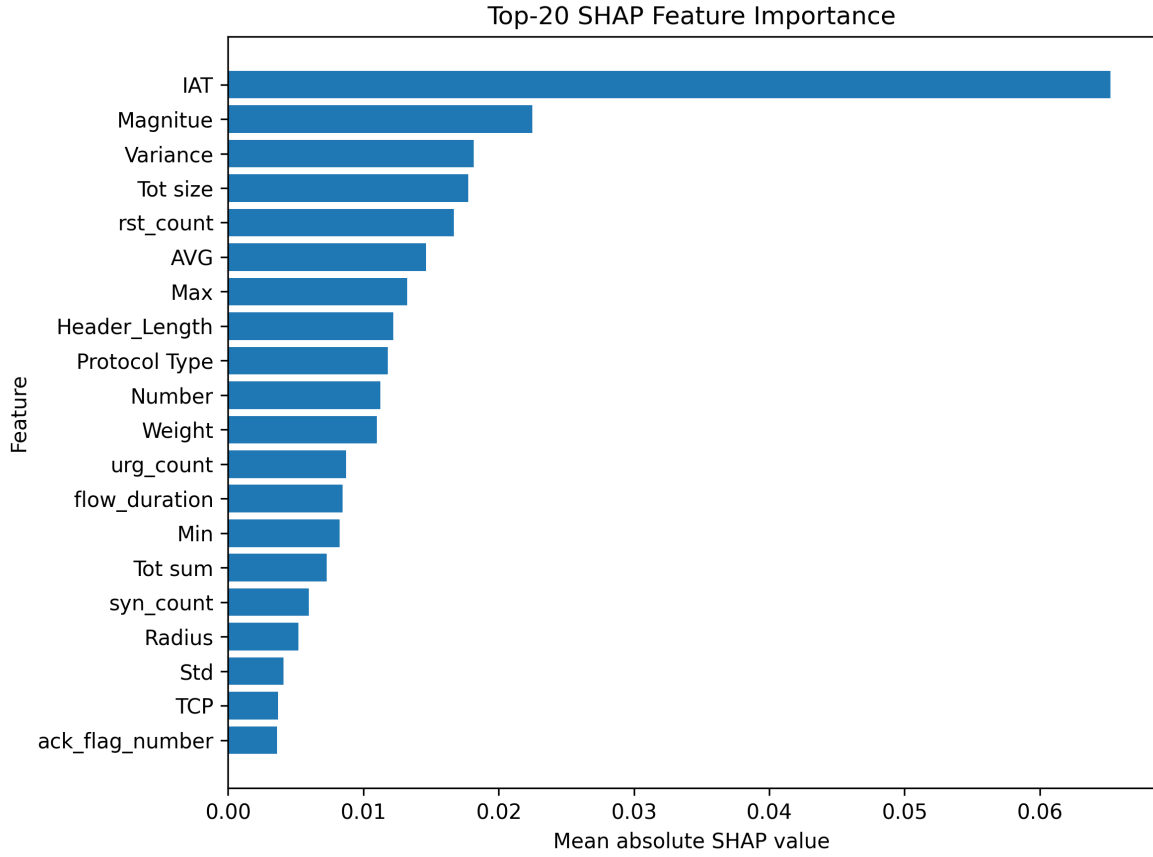


Figure 3: Top-20 SHAP feature importance values for the Random Forest model.

7. Discussion

The results demonstrate that tree-based ensemble models are highly effective for IoT network anomaly detection using tabular flow-level features. Random Forest outperformed both classical boosting models and GPU-trained deep learning models in terms of macro-F1. Although MLP-v2 and 1D-CNN achieved high overall accuracy, their macro-F1 scores were lower than that of Random Forest. This suggests that neural network models may require more advanced class imbalance handling, architectural tuning, or sequence-based traffic representation to outperform ensemble methods on this type of data.

The difference between macro-F1 and weighted-F1 highlights the importance of evaluating intrusion detection models beyond accuracy. In highly imbalanced cybersecurity datasets, dominant classes such as DDoS and DoS can strongly influence weighted metrics, while minority classes such as Brute Force and Web-based attacks remain more difficult to detect. Therefore, macro-F1 and per-class metrics are essential for a realistic evaluation of model performance.

The explainability analysis provides additional insight into model behavior. IAT was consistently identified as the most important feature, indicating that packet inter-arrival timing is highly informative for attack classification. Packet-size statistics and TCP flag-related features also contributed to model decisions. These findings are consistent with the nature of network attacks, which often modify packet frequency, flow duration, packet-size distribution, and connection-state behavior.

8. Conclusions

This study presented a hybrid explainable machine learning and deep learning framework for IoT network anomaly detection using the CICIoT2023 dataset. The framework evaluated binary anomaly

detection and multiclass attack classification tasks under a unified preprocessing and evaluation protocol. Random Forest achieved the best performance among all evaluated models, with an accuracy of 0.9961, macro-F1 score of 0.8698, and weighted-F1 score of 0.9960 for multiclass classification. In the binary anomaly detection task, it achieved an attack F1-score of 0.9988 and ROC-AUC of 0.9998.

The comparison between machine learning and deep learning models showed that tree-based ensemble models were more effective than GPU-trained MLP and 1D-CNN models on tabular network-flow features. Explainability analysis using feature importance, permutation importance, and SHAP values revealed that IAT was the most influential feature, followed by TCP flag-related, temporal, and packet-size statistical features.

Future work will focus on improving minority-class detection using advanced imbalance-aware learning strategies, hybrid neural architectures, and cross-dataset evaluation to assess generalization under unseen IoT attack scenarios.

Funding

This research has been funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. AP25794007).

Data Availability

The CICIOT2023 dataset used in this study is publicly available from the Canadian Institute for Cybersecurity [2].

Acknowledgements

The authors thank the developers of the CICIOT2023 dataset and the open-source machine learning tools used in this study.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

- [1] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, A. A. Ghorbani, CICIOT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment, *Sensors* 23 (2023) 1–20. doi:10.3390/s23135941.
- [2] Canadian Institute for Cybersecurity, CIC IoT Dataset 2023, <https://www.unb.ca/cic/datasets/iotdataset-2023.html>, 2023. Accessed: 2026-06-07.
- [3] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, in: *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, 2018, pp. 108–116. doi:10.5220/0006639801080116.
- [4] N. Moustafa, J. Slay, UNSW-NB15: A comprehensive data set for network intrusion detection systems, in: *2015 Military Communications and Information Systems Conference (MilCIS)*, 2015, pp. 1–6. doi:10.1109/MilCIS.2015.7348942.
- [5] N. Koroniotis, N. Moustafa, E. Sitnikova, B. Turnbull, Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-IoT dataset, *Future Generation Computer Systems* 100 (2019) 779–796. doi:10.1016/j.future.2019.05.041.

- [6] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, A. Anwar, TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems, *IEEE Access* 8 (2020) 165130–165150. doi:10.1109/ACCESS.2020.3022862.
- [7] L. Breiman, Random forests, *Machine Learning* 45 (2001) 5–32. doi:10.1023/A:1010933404324.
- [8] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, *Machine Learning* 63 (2006) 3–42. doi:10.1007/s10994-006-6226-1.
- [9] T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, in: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794. doi:10.1145/2939672.2939785.
- [10] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A highly efficient gradient boosting decision tree, in: *Advances in Neural Information Processing Systems*, volume 30, 2017, pp. 3146–3154.
- [11] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, Y. Elovici, N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders, *IEEE Pervasive Computing* 17 (2018) 12–22. doi:10.1109/MPRV.2018.03367731.
- [12] M. A. Ferrag, L. Maglaras, S. Moschogiannis, H. Janicke, Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study, *Journal of Information Security and Applications* 50 (2020) 102419. doi:10.1016/j.jisa.2019.102419.
- [13] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, M. Guizani, A survey of machine and deep learning methods for internet of things security, *IEEE Communications Surveys & Tutorials* 22 (2020) 1646–1685. doi:10.1109/COMST.2020.2988293.
- [14] Q. A. Al-Haija, A. Droos, A comprehensive survey on deep learning-based intrusion detection systems in internet of things (IoT), *Expert Systems* 42 (2025) e13726. doi:10.1111/exsy.13726.
- [15] F. Ebrahimi, R. Javidan, R. Akbari, Y. Hosseini, Intrusion detection in the internet of things using convolutional neural networks: An explainable AI approach, *Cybersecurity* 8 (2025) 1–22. doi:10.1186/s42400-025-00369-2.
- [16] M. S. Amine, F. A. Nada, K. M. Hosny, Improved model for intrusion detection in the internet of things, *Scientific Reports* 15 (2025) 1–20. doi:10.1038/s41598-025-92852-6.
- [17] D. Gaspar, P. Silva, C. Silva, Explainable AI for intrusion detection systems: LIME and SHAP applicability on multi-layer perceptron, *IEEE Access* 12 (2024) 30164–30175. doi:10.1109/ACCESS.2024.3368377.
- [18] A. Alabbadi, F. Bajaber, An intrusion detection system over the IoT data streams using explainable artificial intelligence (XAI), *Sensors* 25 (2025) 1–25. doi:10.3390/s25030847.
- [19] V. Z. Mohale, I. C. Obagbuwa, Evaluating machine learning-based intrusion detection systems with explainable AI: Enhancing transparency and interpretability, *Frontiers in Computer Science* 7 (2025) 1–23. doi:10.3389/fcomp.2025.1520741.
- [20] H. He, E. A. Garcia, Learning from imbalanced data, *IEEE Transactions on Knowledge and Data Engineering* 21 (2009) 1263–1284. doi:10.1109/TKDE.2008.239.
- [21] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, *Journal of Artificial Intelligence Research* 16 (2002) 321–357. doi:10.1613/jair.953.
- [22] N. Baisholan, M. Turdalyuly, S. Gnatyuk, J. E. Dietz, K. Baisholanova, E. T. Matson, Anomaly detection in imbalanced datasets using FraudX AI, in: *2025 IEEE 13th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Gliwice, Poland, 2025, pp. 1–6. doi:10.1109/IDAACS68557.2025.11322323.
- [23] N. Baisholan, J. E. Dietz, S. Gnatyuk, M. Turdalyuly, E. T. Matson, K. Baisholanova, FraudX AI: An interpretable machine learning framework for credit card fraud detection on imbalanced datasets, *Computers* 14 (2025) 1–30. doi:10.3390/computers14040120.
- [24] M. Zaliskyi, R. Odarchenko, S. Gnatyuk, Y. Petrova, A. Chaplits, Method of traffic monitoring for DDoS attacks detection in e-health systems and networks, *Ceur Workshop Proceedings* 2255 (2018) 193–204.
- [25] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein,

- L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, in: *Advances in Neural Information Processing Systems*, volume 32, 2019, pp. 8024–8035.
- [26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in python, *Journal of Machine Learning Research* 12 (2011) 2825–2830.
- [27] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: *Advances in Neural Information Processing Systems*, volume 30, 2017, pp. 4765–4774.
- [28] A. Fisher, C. Rudin, F. Dominici, All models are wrong, but many are useful: Learning a variable’s importance by studying an entire class of prediction models simultaneously, *Journal of Machine Learning Research* 20 (2019) 1–81.