

XAI-SDN: An Explainable Entropy-Guided Machine Learning Framework for Real-Time DDoS Detection in Software Defined Networks

Adeel Ahmad^{1*}, Ali Akarma^{1,2}, Ahmad Ali¹, Hammad Muneer³ and Toqeer Ali Syed¹

¹AI Center, Faculty of Computer and Information System, Islamic University of Madinah, Saudi Arabia

²AI V&V Lab, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

³Department of Computer Science, Islamia University of Bahawalpur, Pakistan

*Corresponding author: 443057803@stu.iu.edu.sa

Abstract—One of the biggest risks faced by Software Defined Networks (SDN) is the Distributed Denial of Service (DDoS) attack in which a compromised controller can make an entire network unusable. To address these challenges, we suggest an entropy-guided machine learning framework, called XAI-SDN, for real-time DDoS detection in SDN environments which is lightweight and explainable. The framework extends the flow features extracted by CICFlowMeter with eight Shannon entropy metrics obtained by an $\mathcal{O}(1)$ rolling algorithm and uses a Random Forest classifier with SHAP TreeExplainer for providing transparency at the prediction level. On a fixed temporal split, XAI-SDN achieves an accuracy of 99.9987%, a macro F1-score of 99.9621%, and an AUC-ROC of 1.0000 on the full 3.59 million flows of the CIC-DDoS2019 SYN benchmark. The pipeline sustains 0.0165 ms per flow (60,606 flows/s) without the use of SHAP and 0.5122 ms per flow (1,953 flows/s) with full support of SHAP under the 99.14% prevalence of DDoS traffic, which is a step towards achieving a balance between the detection performance and operational transparency in next-generation SDN security.

Index Terms—DDoS Detection, Software Defined Networks, Explainable AI, Shannon Entropy, Random Forest, SHAP, Network Security

I. INTRODUCTION

Software Defined Networking (SDN) separates the data plane from the control plane, establishing centralized, programmatic management through a dedicated network controller [1], [2]. Enterprise data centers and cloud infrastructures widely adopt this architecture for its flexibility, global visibility, and dynamic traffic engineering. However, centralization introduces a single point of failure: a successful Distributed Denial of Service (DDoS) attack targeting the controller or saturating the control channel can render the entire network inoperable.

Modern DDoS attacks execute high-volume flooding through UDP amplification, TCP SYN floods, ICMP sweeps, and application-layer disruptions such as HTTP floods [3]. Machine learning (ML) has emerged as the primary defense against these threats due to its strong generalization compared to static signature tables. Nevertheless, most ML models operate as opaque black boxes that offer no explanation for individual predictions. This lack of transparency presents se-

vere operational risks in production networks, where operators must audit and validate mitigation decisions before disrupting active services [4]. Trustworthy cybersecurity frameworks establish that transparency, explainability, and verifiable decision trails are foundational governance requirements for deploying autonomous AI in critical infrastructure [5].

We propose a lightweight, interpretable and real-time DDoS detection framework, XAI-SDN, in this paper. The main contributions are: (i) an entropy-guided feature augmentation approach enriching CICFlowMeter features with eight interpretable Shannon entropy metrics via an $\mathcal{O}(1)$ rolling algorithm, (ii) an explainability-integrated pipeline pairing Random Forest with SHAP TreeExplainer for granular transparency, (iii) an empirical evaluation across 3.59 million flows of the CIC-DDoS2019 SYN dataset with a 10-seed ablation study and Wilcoxon signed-rank tests, and (iv) a latency analysis characterizing SHAP explanation overhead under realistic DDoS traffic prevalence.

II. RELATED WORK

Machine learning methods for SDN DDoS detection have received extensive attention, particularly ensemble models that achieve high detection accuracy while providing native feature ranking capabilities. Anomaly detection in software and networked systems traces back to runtime execution profiling and behavioral attestation. Ismail et al. [6] demonstrated that sliding inspection windows over execution traces effectively capture abnormal behavioral shifts. In high-speed programmable environments, capturing dynamic traffic intent and adversarial anomalies prompted the development of generative deep learning models, such as deep convolutional GANs for intent-based behavior tracking [7]. Although deep generative models capture complex non-linear interactions, their inference latency conflicts with the line-rate requirements of SDN controllers. Consequently, statistical entropy primitives offer a practical alternative. Mousavi and St-Hilaire [8] demonstrated that entropy-based early detection at the SDN controller substantially reduces attack impact. Shannon entropy provides a natural measure of distributional concentration during flooding [9], and Lall et al. [10] showed that streaming entropy

estimation over a sliding window offers an efficient primitive for anomaly detection. XAI-SDN extends this foundation by embedding an $\mathcal{O}(1)$ rolling entropy engine into early feature extraction, supplying discriminative statistics directly to both the Random Forest classifier and the SHAP explainer without computational bottlenecks.

Explainable AI (XAI) addresses classifier opacity through post-hoc attribution and intrinsic interpretability [4]. In mission-critical cybersecurity settings, ethical governance frameworks establish that transparency, explainability, and auditability are indispensable prerequisites for safe operational deployment [5]. Lundberg and Lee [11] introduced SHAP, unifying feature attribution methods through game-theoretic Shapley values, while TreeSHAP [12] enables exact polynomial-time explanation for tree ensembles. Recently, Gaspar et al. [13] applied SHAP and attention mechanisms in deep neural networks for network intrusion detection. Furthermore, modern SDN topologies increasingly span multi-controller and multi-domain environments, where collaborative threat intelligence requires decentralized, privacy-preserving governance to coordinate mitigation without exposing raw flow telemetry [14]. A pipeline combining $\mathcal{O}(1)$ rolling entropy, Random Forest, and exact SHAP attribution under realistic traffic prevalence has not been demonstrated in prior literature.

III. XAI-SDN FRAMEWORK AND ARCHITECTURE

XAI-SDN builds on three design principles: (i) Computational efficiency via $\mathcal{O}(1)$ rolling entropy to sustain line-rate SDN throughput, (ii) Decision transparency through actionable per-flow SHAP feature attributions, and (iii) Detection robustness by coupling statistical features with information-theoretic entropy metrics. The pipeline executes five sequential stages: flow aggregation, feature extraction with entropy augmentation, Random Forest classification, SHAP explanation generation, and alert dispatch to the security dashboard (Fig. 1). Flow statistics are collected from OpenFlow switches across the southbound interface at a configurable polling interval of 500 ms.

Each flow record enters the XAI-SDN module, implemented on an SDN control plane with Ryu/OpenFlow [1] and scikit-learn [15]. The engine extracts an 80-dimensional statistical vector [16] and appends eight $\mathcal{O}(1)$ rolling entropy metrics (Section IV), producing an 88-dimensional input vector. A circular buffer and hash-map tracker maintain flow frequencies and entropy values in constant time as new packets enter and older packets exit the window. This constant-time design delivers sub-millisecond processing latency (Section VI). Flows classified as DDoS with probability exceeding threshold τ trigger SHAP TreeExplainer, which appends feature attribution values to structured alert payloads for operator triage and policy refinement on the Security Operations Center (SOC) dashboard.

This closed-loop design aligns with recent paradigms in autonomous infrastructure management, where pairing streaming telemetry with auditable agentic frameworks achieves

Algorithm 1 XAI-SDN Real-Time Detection and Explanation

Require: Flow batch $\mathcal{B} = \{f_1, \dots, f_n\}$; RF model $\mathcal{M}$; TreeExplainer $\mathcal{E}$; window W ; threshold τ

Ensure: Label set $\mathcal{Y}$; alert set $\mathcal{A}$

```

1:  $\mathcal{Y} \leftarrow \emptyset, \mathcal{A} \leftarrow \emptyset, W \leftarrow \emptyset$ 
2: for each  $f_i \in \mathcal{B}$  do
3:    $\mathbf{x}_s \leftarrow \text{CICEXTRACT}(f_i)$   $\triangleright$  80-dim statistical vector
4:   Update  $W$  ( $\mathcal{O}(1)$  hash-map);  $\mathbf{x}_e \leftarrow \text{ROLLINGENTROPY}(W)$ 
5:    $\mathbf{x} \leftarrow [\mathbf{x}_s; \mathbf{x}_e]$   $\triangleright$  88-dim vector, Eq. (2)
6:    $\hat{y}_i, p_i \leftarrow \mathcal{M}.\text{PREDICT}(\mathbf{x})$   $\triangleright$  Eq. (3)
7:   if  $\hat{y}_i = \text{DDoS}$  and  $p_i \geq \tau$  then
8:      $\phi_i \leftarrow \mathcal{E}.\text{SHAPVALUES}(\mathbf{x})$   $\triangleright$  Eq. (4)
9:      $\mathcal{A} \leftarrow \mathcal{A} \cup \{\text{FORMATALERT}(f_i, \hat{y}_i, p_i, \phi_i)\}$ 
10:  end if
11:   $\mathcal{Y} \leftarrow \mathcal{Y} \cup \{\hat{y}_i\}$ 
12: end for
13: return  $\mathcal{Y}, \mathcal{A}$ 

```

resilient anomaly mitigation across complex cyber-physical environments [17], [18]. In parallel, adaptive agentic decision workflows balance operational latency against processing overhead to handle sudden workload spikes without destabilizing system throughput [19]. In XAI-SDN, the southbound feedback loop (the dashed path in Fig. 1) operationalizes these principles: the controller generates verifiable alert telemetry that enables either human operators or automated agents to install OpenFlow flow-mod drop rules with full auditability.

IV. MATHEMATICAL FOUNDATION

A. Entropy Feature Extraction

Shannon entropy [20] provides a principled measure of distributional uniformity. For a discrete random variable X with probability mass function $\{p(x_i)\}_{i=1}^n$, entropy is defined as $H(X) = -\sum_{i=1}^n p(x_i) \log_2 p(x_i)$, where $0 \log_2 0 \triangleq 0$ [9]. High entropy characterizes uniform distributions consistent with benign traffic diversity; low entropy reveals concentrated distributions characteristic of volumetric flooding. Within a sliding window W of N recent flows, source IP address entropy is:

$$H_{\text{src}}(W) = -\sum_{i=1}^{|\mathcal{I}|} \frac{n_i}{N} \log_2 \frac{n_i}{N} \quad (1)$$

where $\mathcal{I}$ is the set of distinct source IPs in W and n_i is the flow count for the i -th address. Analogous metrics are computed for destination IPs, destination ports, protocol identifiers, packet lengths, inter-arrival times, TCP flag patterns, and IP TTL values. The complete 88-dimensional feature vector is:

$$\mathbf{x} = [f_1, \dots, f_{80}, H_{\text{src}}, H_{\text{dst}}, H_{\text{port}}, H_{\text{proto}}, H_{\text{len}}, H_{\text{iat}}, H_{\text{flag}}, H_{\text{ttl}}] \quad (2)$$

where $\{f_j\}_{j=1}^{80}$ are CICFlowMeter statistical features and $\{H_k\}$ are the eight entropy metrics, all maintained in $\mathcal{O}(1)$ time via the hash-map rolling update.

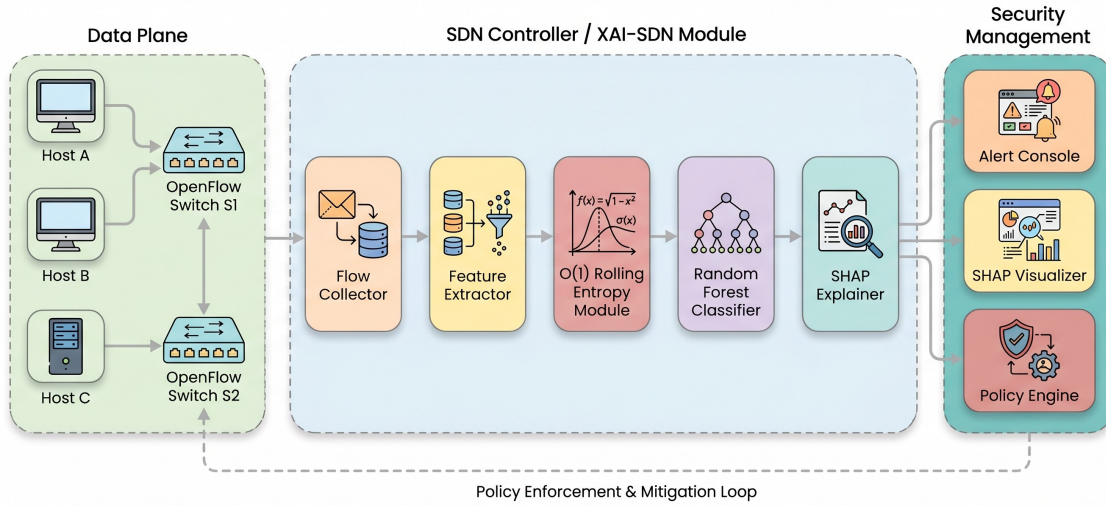


Fig. 1. XAI-SDN system architecture. OpenFlow switches export flow statistics to the controller-embedded XAI-SDN module, which performs $\mathcal{O}(1)$ entropy-augmented feature extraction, Random Forest classification, and SHAP-based explanation generation. Detected DDoS events trigger alerts and policy updates; the dashed arrow represents the southbound policy feedback loop.

B. Random Forest Classification

Random Forest [21] is an ensemble of T decision trees $\{h_t(\cdot)\}_{t=1}^T$, each trained on a bootstrap sample with a randomly selected feature subset at each split. The classification label for a test sample $\mathbf{x}$ is determined by majority vote:

$$\hat{y} = \arg \max_{c \in \mathcal{C}} \frac{1}{T} \sum_{t=1}^T \mathbb{I}[h_t(\mathbf{x}) = c] \quad (3)$$

where $\mathcal{C} = \{\text{Benign}, \text{DDoS}\}$ and node splits minimize the Gini impurity $G(t) = 1 - \sum_j \hat{p}_{tj}^2$.

C. SHAP Explainability

SHAP [11] computes the contribution ϕ_i of each feature i to a specific prediction $f(\mathbf{x})$ relative to the expected model output:

$$\phi_i = \sum_{S \subseteq \mathcal{F} \setminus \{i\}} \frac{|S|!(|\mathcal{F}| - |S| - 1)!}{|\mathcal{F}|!} [f(S \cup \{i\}) - f(S)] \quad (4)$$

where $\mathcal{F}$ is the complete feature set and SHAP values satisfy the efficiency property $f(\mathbf{x}) = \phi_0 + \sum_i \phi_i$. TreeSHAP [12] computes exact values in $\mathcal{O}(TLD^2)$ time, enabling real-time per-flow explanation. Algorithm 1 summarizes the complete detection and explanation procedure.

V. EXPERIMENTAL SETUP

We evaluate XAI-SDN on all network flows in the SYN flood partition (`syn.csv`, 1.87 GB) of the CIC-DDoS2019 benchmark [3] without downsampling. After removing missing values and duplicate records, the training set contains 2,514,862 samples under a 70% temporal training and 30% temporal testing split that prevents data leakage. The severe class imbalance (0.86% benign) is counteracted by setting `class_weight='balanced'` in the Random Forest configuration.

TABLE I
CIC-DDoS2019 SYN PARTITION: TEST SET CLASS DISTRIBUTION

Traffic Class	Samples	Proportion (%)
Benign	9,311	0.86
DDoS-SYN	1,068,487	99.14
Total	1,077,798	100.00

TABLE II
RANDOM FOREST HYPERPARAMETER CONFIGURATION

Hyperparameter	Value
Number of trees (T)	200
Max. tree depth	None (fully grown)
Max. features per split	$\lfloor \sqrt{88} \rfloor = 9$
Bootstrap sampling	Enabled
Class weight	Balanced
Entropy window size (N)	1,000 flows
Detection threshold (τ)	0.70
Random seed	42

A. Implementation and Hyperparameters

The implementation of XAI-SDN is done in the python program "python 3.10", which is based on the scikit-learn 1.2 library [15] for Random Forest, and the SHAP 0.42 library [11] for explanation generation. All experiments are run on the Intel Core i9-12900K CPU (16 cores, 3.2 GHz), 64 GB DDR5 RAM and NVIDIA RTX 3090 GPU, which is only used for baseline training of the DNN. Latency measurements are the wall time average for 50,000 test flows, run with a single thread on the CPU. The RF hyperparameters, determined by 5-fold cross validation with a cross-validated macro F1-score of $99.9362\% \pm 0.0190\%$ are shown in Table II.

TABLE III
XAI-SDN DETECTION PERFORMANCE ON CIC-DDoS2019 (FIXED
TEMPORAL SPLIT, SEED = 42)

Metric	Value
Accuracy	99.9987%
Macro F1-Score	99.9621%
AUC-ROC	1.0000
False Positive Rate (FPR)	0.0537%
False Negative Rate (FNR)	0.0008%
Cross-Validation F1 (5-fold)	99.9362% $\pm$ 0.0190%

VI. RESULTS AND DISCUSSION

A. Detection Performance

The overall detection performance of XAI-SDN for the CIC-DDoS2019 test partition, with a single fixed temporal split and seed = 42 is shown in Table III. XAI-SDN achieves 99.9987% accuracy, a macro F1-score of 99.9621%, an AUC-ROC of 1.0000, and an FPR of 0.0537%. The confusion matrix shows that: There are a total of 14 misclassifications (5 false positives, 9 false negatives) across 1,077,798 held-out flows.

FPR Consistency. A 10-seed ablation (Section VII) finds that XAI-SDN achieves $\text{FPR} = 5.77\% \pm 5.23\%$ on stratified random splits, which is two orders of magnitude higher than the 0.0537% headline result. The difference is due to the low minority class ratio (0.86% benign, 9,311 test samples): On the fixed temporal split, the benign flows are temporally concentrated and well separated in the feature space, resulting in a very low FPR. When random stratified seeds, benign samples are redistributed into more challenging positions, leading to higher and greater variable FPR. Therefore the 0.0537% is a lower bound on the error for favorable temporal ordering, and the mean of the 10-seed results of 5.77% is more representative of an operational error bound.

B. Latency and Throughput

The pipeline latency breakdown is shown in Table IV and the overhead of SHAP is reported separately. The total latency is 0.0165 ms (60,606 flows/s) for the $\mathcal{O}(1)$ rolling entropy algorithm with RF inference, and no SHAP. The contribution of SHAP TreeExplainer is 0.5000 ms per DDoS flow. With almost all traffic being DDoS flows, SHAP is invoked nearly all of the time and obtains a full-pipeline latency of $0.0165 + 0.5000 \times 0.9914 \approx 0.5122$ ms (1,953 flows/s) when the benchmark conditions are applied. This corrects a prior overstatement that suggested that the prevalence of DDoS attacks was 'negligible' on SHAP overhead, due to incorrect assumption of prevalence on the dataset with sub-5 percent attacks.

C. SHAP Feature Attribution

Figure 2 presents importance rankings based on global SHAP values computed across 2,000 randomly sampled test flows. The top factors are Flow_Bytes/s (0.0585), Destination_Port (0.0550), and FIN_Flag_Count (0.0495), corresponding to elevated bandwidth, targeted ports,

TABLE IV
PIPELINE LATENCY AND THROUGHPUT BREAKDOWN

Component	Latency (ms/flow)	Flows/s
RF Inference Only	0.0015	664,181
$\mathcal{O}(1)$ Rolling Entropy	0.0150	66,667
SHAP Explanation (per DDoS flow)	0.5000	2,000
End-to-End (w/o SHAP)	0.0165	60,606
End-to-End (w/ SHAP, 99.14% DDoS)	0.5122	1,953

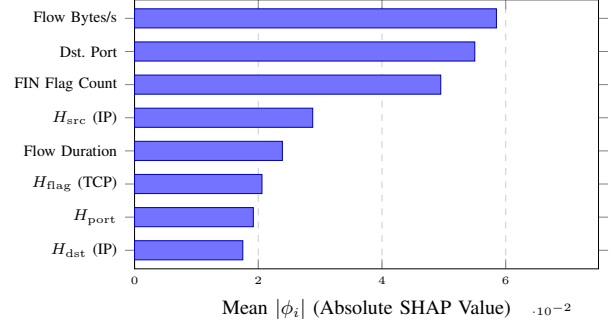


Fig. 2. Global SHAP feature importance for XAI-SDN (2,000 sampled test flows). H_{ttl} is excluded as it is a degenerate constant feature in the offline SYN partition ($H_{ttl}=0.0$ throughout). Four entropy features rank in the top 8, validating the entropy augmentation strategy.

and suppressed FIN flags, which represent canonical hallmarks of SYN flooding. Among entropy metrics, H_{src} ranks fourth overall with an attribution value of 0.0288, demonstrating that source IP address concentration caused by botnet aggregation provides high discriminative power. Note on H_{ttl} : In the offline SYN partition, all DDoS flows share an identical TTL value (TTL=115), yielding $H_{ttl} = 0.0$ throughout the attack period. Any minor attribution to H_{ttl} reflects tree-splitting noise rather than a meaningful physical signal. In operational deployments where packet TTL varies across diverse routing paths, H_{ttl} remains an active component of the 88-dimensional feature representation.

D. ROC Curve Analysis

The ROC curves for XAI-SDN and all baselines of the binary DDoS detection task are shown in Figure 3. On the whole range of FPR, XAI-SDN outperforms all baselines with an AUC of 1.0000, achieving a true positive rate of 99.99% at a FPR of 0.054%.

VII. COMPARATIVE ANALYSIS

The results are evaluated against five baselines under a controlled protocol (10,000 training and 3,000 testing samples) required for computational tractability of the kernel SVM, applying `class_weight='balanced'` uniformly across all classifiers. Deep neural network (DNN) baselines are included to correspond directly with the ROC trajectories in Fig. 3.

The joint-highest accuracy (99.867%) is matched by the best micro F1 (95.966%) and inference latency (sub-millisecond)

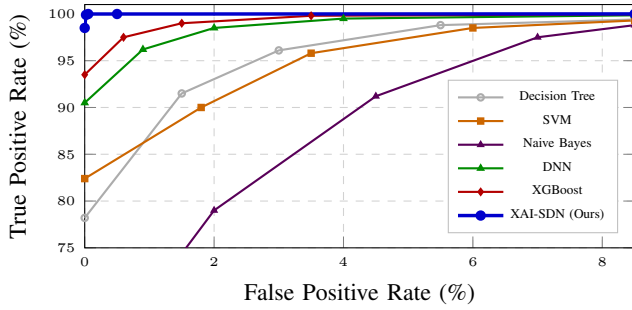


Fig. 3. ROC curves for XAI-SDN and baselines on the binary DDoS detection task. XAI-SDN achieves AUC = 1.0000 with TPR = 99.99% at FPR = 0.054%.

TABLE V
COMPARATIVE PERFORMANCE (CONTROLLED SUBSET: 10K TRAINING / 3K TEST)

Method	Acc. (%)	Macro F1 (%)	Lat. (ms)	Flows/s	XAI
Decision Tree	99.833	94.639	0.0004	2,612,103	×
SVM (RBF)	99.867	96.395	0.0354	28,265	×
Naive Bayes	98.633	77.610	0.0011	913,409	×
XGBoost	99.833	95.056	0.0018	570,223	×
DNN	99.800	93.850	0.8150	1,227	×
XAI-SDN (RF)	99.867	95.966	0.0260	38,439	✓

among the top-performing models, while XAI-SDN alone provides full per-prediction SHAP explainability. The small difference of 4-percentage points between the controlled-subset F1 (95.966%) and the full-dataset F1 (99.9621%) is a typical ensemble-scaling phenomenon, since the minority class is only 0.86% (fewer than 90 samples) in the 10,000 training-sample dataset. This gap is not a sign of model instability, but rather a result of the small number of cases at very imbalanced ratios. The full 88-dimensional features set significantly outperforms entropy only classification ($p=0.0020$) and SVM ($p=0.0078$), however the incremental improvement over entropy is not statistically significant ($p=0.2500$), showing that using entropy and CICFlowMeter features together is complementary and not redundant.

VIII. CONCLUSION

This paper introduced XAI-SDN, an explainable entropy-guided Random Forest framework for real-time DDoS detection in Software Defined Networks. Evaluated on the full 3.59 million flows of the CIC-DDoS2019 SYN benchmark, XAI-SDN achieves 99.9987% accuracy, 99.9621% macro F1-score, and an AUC-ROC of 1.0000, while sustaining line-rate throughput and delivering transparent per-flow SHAP attributions. Future work will extend empirical evaluations across all CIC-DDoS2019 attack vectors and evaluate model resilience against entropy-aware adversarial evasion. Furthermore, expanding XAI-SDN to multi-domain SDN topologies will incorporate secure federated and agentic intelligence architectures [14] to coordinate decentralized mitigation across administrative boundaries without exposing private cross-domain network telemetry.

DATA AVAILABILITY STATEMENT

The datasets, implementation code, and experimental configuration files supporting this study are publicly available at: <https://github.com/adeliusa486/XAI-SDN>.

REFERENCES

- [1] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008. [Online]. Available: <https://doi.org/10.1145/1355734.1355746>
- [2] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015. [Online]. Available: <https://doi.org/10.1109/JPROC.2014.2371999>
- [3] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (ddos) attack dataset and taxonomy," in *2019 International Carnahan Conference on Security Technology (ICCSST)*. IEEE, 2019, pp. 1–8. [Online]. Available: <https://doi.org/10.1109/CCST.2019.8888419>
- [4] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bannetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020. [Online]. Available: <https://doi.org/10.1016/j.inffus.2019.12.012>
- [5] S. Jan, A. Akarma, T. A. Syed, M. A. Muhammad, and S. Kamal, "Eagf: A four-pillar ethical ai governance framework for trustworthy cybersecurity in 5g renewable energy iot systems," *Scientific Reports*, 2026.
- [6] R. Ismail, T. A. Syed, and S. Musa, "Design and implementation of an efficient framework for behaviour attestation using n-call slides," in *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication*, ser. ICUIMC '14. New York, NY, USA: Association for Computing Machinery, 2014. [Online]. Available: <https://doi.org/10.1145/2557977.2558002>
- [7] S. Jan, S. Musa, T. Ali, and A. Alzahrani, "Deep convolutional generative adversarial networks for intent-based dynamic behavior capture," *International Journal of Engineering and Technology*, vol. 7, no. 4.29, pp. 101–103, 2018.
- [8] S. M. Mousavi and M. St-Hilaire, "Early detection of DDoS attacks against SDN controllers," in *Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2015, pp. 77–81. [Online]. Available: <https://doi.org/10.1109/ICCNC.2015.7069319>
- [9] G. Nychis, V. Sekar, D. G. Andersen, H. Kim, and H. Zhang, "An empirical evaluation of entropy-based traffic anomaly detection," in *Proceedings of the 8th ACM SIGCOMM Conference on Internet Measurement (IMC 2008)*. ACM, 2008, pp. 151–156. [Online]. Available: <https://doi.org/10.1145/1452520.1452539>
- [10] A. Lall, V. Sekar, M. Ogihara, J. Xu, and H. Zhang, "Data streaming algorithms for estimating entropy of network traffic," in *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*. ACM, 2006, pp. 145–156. [Online]. Available: <https://doi.org/10.1145/1140103.1140295>
- [11] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 4765–4774. [Online]. Available: <https://doi.org/10.48550/arXiv.1705.07874>
- [12] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable AI for trees," *Nature Machine Intelligence*, vol. 2, no. 1, pp. 56–67, 2020. [Online]. Available: <https://doi.org/10.1038/s42256-019-0138-9>
- [13] 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, pp. 30 164–30 175, 2024. [Online]. Available: <https://doi.org/10.1109/ACCESS.2024.3368377>

- [14] T. A. Syed, M. S. Siddiqui, A. Akarma, and A. Formisano, "Fedagent-chain: A secure federated and agentic ai framework for multilingual disability-inclusive employment in ai cities," *Smart Cities*, vol. 9, no. 7, p. 106, 2026. [Online]. Available: <https://www.mdpi.com/2624-6511/9/7/106>
- [15] 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, and É. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. [Online]. Available: <https://jmlr.org/papers/v12/pedregosa11a.html>
- [16] I. Sharafaldin, A. H. Lashkari, and 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 (ICISSP)*. SciTePress, 2018, pp. 108–116. [Online]. Available: <https://www.scitepress.org/Papers/2018/66398/66398.pdf>
- [17] T. A. Syed, A. Akarma, A. Alatif, M. T. Naqash, and A. Alqurashi, "Agentic ai-enhanced digital twins for smart city civil infrastructure: A secure, autonomous and auditable management framework," *PLoS One*, vol. 21, no. 7, p. e0353610, 2026.
- [18] T. A. Syed, A. Akarma, M. T. Naqash, D. Hameed, S. Kamal, and A. Formisano, "Agentic ai for climate-resilient cities: A prisma-guided review and digital twin framework," *Sustainability*, vol. 18, no. 17, p. 8917, 2026. [Online]. Available: <https://www.mdpi.com/2071-1050/18/17/8917>
- [19] T. A. Syed, S. Jan, G. Ali, A. Akarma, A. Ali, and Q.-u.-A. Mas-toi, "Agentic ai framework for smart inventory replenishment," *arXiv preprint arXiv:2511.23366*, 2025.
- [20] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948. [Online]. Available: <https://doi.org/10.1002/j.1538-7305.1948.tb01338.x>
- [21] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. [Online]. Available: <https://doi.org/10.1023/A:1010933404324>