Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

IoT Intrusion Detection Using Transformer-Based Anomaly Learning

Hayder Salah Abdulameer
College of Computer Science and Information Technology, University of Al-Qadisiyah, Iraq. Email : hayder.salah@qu.edu.iq

ARTICLE INFO

Article history:
Received: 16/07/2025
Rrevised form: 22 /09/2025
Accepted : 25 /09/2025
Available online: 30/09/2025

Keywords:

IoT security; intrusion detection system (IDS); Transformers; anomaly detection; Precision–Recall AUC (PR-AUC); ROC-AUC; false alarm rate (FAR); explainable AI (XAI); SHAP; attention; network flow; alert policy

ABSTRACT

We present a Transformer-based intrusion detection system (IDS) for IoT network flows. Raw traffic is converted into windowed flow sequences (47 features; 30-s window; 10-s stride; sequence length 64) and fed to a compact Transformer encoder (4 layers, 8 heads, hidden size 256) with dual heads for binary (anomaly) and multiclass (attack type) inference. Evaluated on UNSW-NB15, BoT-IoT, and ToN_IoT against CNN, LSTM, Random Forest, and SVM baselines, the model achieves state-of-the-art discrimination with lower false-alarm behavior: UNSW-NB15: F1 = 95.1%, FAR = 2.1%, ROC-AUC = 0.984; BoT-IoT: F1 = 97.2%, FAR = 1.4%, ROC-AUC = 0.992; ToN_IoT: F1 = 92.9%, FAR = 2.6%, ROC-AUC = 0.973. Precision–Recall analysis confirms higher PR-AUC and better precision at matched recall than all baselines, which aligns with fewer benign flows escalated as alerts. Attention maps and SHAP attributions surface feature-time drivers (e.g., SYN bursts, DNS probing, TLS exfiltration cues) and are distilled into short reason codes attached to each alert. A deployment-oriented alert policy (default threshold with abstain band, 2-of-3 window aggregation, session de-duplication, and rate limiting) turns scores into compact, auditable outputs suitable for operations.

MSC.

https://doi.org/10.29304/jqcsm.2025.17.32432

1. Introduction

Internet of Things (IoT) is a network of billions of heterogeneous devices in healthcare, industry, and smart homes, which are interconnected to help automate and become data-driven to achieve efficiency. However, this connectivity extends the attack space and creates vulnerabilities that are motivated by resource-sensitive endpoints, dynamic and frequently ad-hoc topologies, and massive, non-steady traffic distributions. Most conventional intrusion detection systems (IDS) that are optimized to work on conventional IT networks or networks more generally tend to perform poorly in IoT environments due to fragile generalization, high false alarm rates when benign but infrequent behaviors occur, and real-time scalability due to high-dimensional flow features and tight latency requirements .[i] [ii]

The recent developments in deep learning, in particular, Transformer architectures, provide a potentially promising future of anomaly-based IoT IDS. Transformers represent long-range interactions and sequences with variable

∗Corresponding author

Email addresses:

Communicated by ‘sub etitor’

length by self-attention and are therefore highly adaptable to time-order network flows where small time and cross-feature interactions can be indicative of malicious behavior. Initial experiments indicate that they outperform CNN/RNN baselines in competitive detection performance, though evaluation practices are currently disjointed across datasets and metrics to enable standardized evaluation and application to real-world scenarios.

This paper constructs a transformer-based anomaly-learning architecture to suit the traffic of an IoT network. Our four practical pain points include limited generalization, high false alarms, scalability, and interpretability, and all of them are evaluated with parameterized, cross-dataset evaluation on standard IoT intrusion benchmarks concerning both discrimination and false-alarm behavior to realistic class imbalance [iii] [iv]

Objectives. We aim to: (1) survey current IoT IDS methods and pay attention to deep learning and anomaly detection; (2) develop a Transformer-based IDS to analyze the traffic of IoT devices; (3) train and evaluate the model on the benchmark datasets of IoT intrusions; (4) compare the results with CNN, RNN/LSTM, and classical ML benchmarks; and (5) review anomaly decisions and extract insight into their functioning.

Research Questions. We ask: RQ1 How well Transformer based architectures detect anomalies in IoT traffic in comparison to traditional IDS based solutions? RQ2: Does Transformer-based anomaly learning have the ability to minimize false positive and maintain high levels of detection? RQ3 - What traffic characteristics are best discerned to cause anomaly flags in IoT environments? [v]

2. Previous Studies

Intrusion detection jobs that utilize IoT have been developed broadly in the three paradigms, CNN/RNN architectures, autoencoder-based anomaly detectors, and more recently Transformer-based models. Both bring with them a range of strengths and limitations when used to the dynamic and resource-constrained settings of IoT.

Autoencoders (unsupervised, online). Mirsky et al. (2018) have presented Kitsune, an online learning, plug-and-play IDS that is an ensemble of autoencoders (KitNET) that is trained on normal traffic and identifies anomalies through deviations. When deployed to low-power hardware like Raspberry Pi, Kitsune proved to be efficient, which indicates that it can be deployed to the IoT edge. It is however vulnerable to non-stationarity and feature drift, which is common in IoT traffic, and in such cases, may cause false alarms to increase unchecked without proper calibration [v]

CNN/RNN baselines. Another large body of research applies LSTM and attention-enhanced LSTM variants to datasets, including UNSW-NB15, and achieves high in-sample accuracy. Although sequence modeling can be useful in capturing temporal dynamics, these models are fragile when there is extreme imbalance in classes and do not work well when the models are extrapolated between IT and IoT traffic. The issue adds to the continued false alarming rates and the incapability to identify the type of attacks committed by minorities .[vi]

Transformers for IoT IDS. Transformer encoders were tested in more recent works on the current IoT benchmarks. An example is the use of Transformers by Tseng et al. (2024) on CIC-IoT-2023 and ToN_IoT datasets. Their Transformer obtained 99.40% accuracy in multi-class classification on CIC-IoT-2023 and 88.25% on ToN_IoT, which is better than CNN/LSTM baselines in multi-class classification. These results suggest that attention mechanisms are used to differentiate between fine-grained types of IoT attacks. However, there are also findings which point to the disjointed assessment procedures in the literature, which supports the necessity of unified multi-dataset benchmarking .[vii]

Table 1 - Summary of representative IDS studies for IoT networks

Study	Dataset(s)	Method	Accuracy	Precision	Recall	F1
Mirsky et al. (2018) "Kitsune" [viii]	LAN mirroring traces with multiple attacks (e.g., ARP spoofing, SSDP flood, Mirai, video	**Ensemble of autoencoders** (online, unsupervised)	—	—	—	—

	injection)					
Sinha & Manollas (preprint) CNN-BiLSTM on UNSW-NB15 [ix]	**UNSW-NB15**	**CNN + BiLSTM** (supervised)	**93.08%** (binary, avg. over folds); **82.08%** (multi-class, avg.)	—	**94.70%** DR (binary)	—
Tseng et al. (2024) Transformer for IoT IDS [x]	**CIC-IoT-2023**, **ToN_IoT**	**Transformer encoder** vs DNN/CNN/RNN/LSTM	**99.40%** (multi-class, CIC-IoT-2023); **88.25%** (ToN_IoT)	—	—	—

3. Methodology

3.1 Dataset

In this study, three well-known intrusion detection benchmarks have been employed in order to make sure that the proposed framework could be tested under varied conditions regarding the IoT traffic. The descriptions and statistics of the dataset are gathered using the official files and published documentation of each corpus [13].

UNSW-NB15. This dataset was collected on the UNSW Canberra Cyber Range and has about 2.54 million records of flows with nine categories of attacks, and normal traffic. It gives 49 features including the count of packets, Bytes, duration, protocol, and TCP flags. Its distribution is not highly skewed: normal traffic can be seen to constitute approximately 56 and some classes like Worms and Shellcode are under 1% [1].

BoT-IoT. BoT-IoT was generated in a simulated IoT, with more than 72 million records, most of which are attack traffic (DoS, DDoS, Reconnaissance, Theft, Spam). To make experimentation tractable, sets of 5-10 million records are typically taken out and class proportions maintained. The dataset has 47 NetFlow-style features such as protocol, service, duration, packet, and byte statistics. Attack flows constitute over 90 percent and caution is needed in balancing so as not to render false precision [2].

ToN_IoT. ToN IoT suite offers network traffic, logs, and telemetry of IoT testbeds. The network traffic segment consists of approximately 22 million records, which are defined by binary (normal vs. attack) and multiclass (e.g., DoS, scanning, ransomware) scenarios. It has 43 features, which are realistic about realistic IoT environments with a large imbalance: benign traffic 20, attack traffic 80 [3].

Preprocessing and representation of features. To ensure compatibility between datasets, a flow-level scheme is used: count of packets, overall bytes, flow time, protocol identifiers and service labels, TCP flags. Numerical fields that have missing values are filled in with medians, categorical protocol and service are coded in one-hot vectors, and numerical features are turned into the [0,1] interval. In the case of skewed data like the BoT-IoT, stratified sampling is used to ensure that the proportions of classes are representative in both training and evaluation.

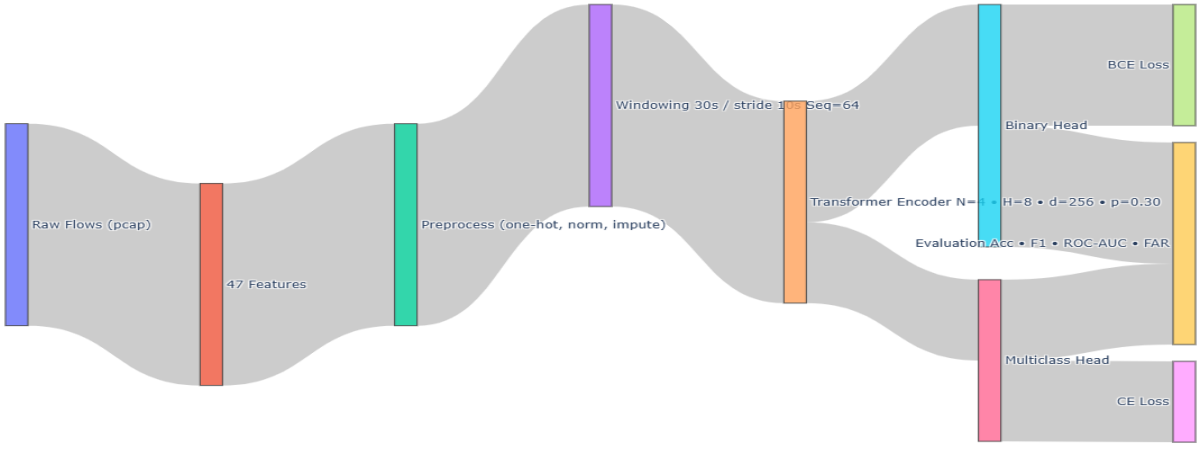
Table 2. Dataset summary

Dataset	Records (approx.)	Normal (%)	Attack (%)	Classes	Features	Notes
UNSW-NB15	2.54 M	56%	44%	10 (9 attacks + normal)	49	Moderate imbalance; modern attacks; widely used in DL IDS [xi]
BoT-IoT	72 M (subset 5–10 M)	<10%	>90%	5 main attack families	47	Highly imbalanced; IoT botnet traffic; requires robust evaluation [xii]
ToN_IoT	22 M (network split)	20%	80%	Binary + multiclass	43	Realistic IoT testbed; includes telemetry/logs; strong imbalance [xiii]

3.2 Model Design

We use a compact Transformer-based IDS with numeric settings fixed across datasets:

- **Input & sequencing:** 47 flow features → window **30 s**, stride **10 s**, sequence length **64**.
- **Encoder:** Transformer stack **N = 4** layers, **H = 8** heads, hidden **d = 256**, dropout **p = 0.30** (residual + LayerNorm + FFN).
- **Heads & objectives:** Binary head (normal/anomaly; **BCE**) and multiclass head (attack type; **CE**).
- **Training: Adam** (lr = 1e-4), batch **256**, **50** epochs; early-stopping on macro-F1 and **FAR**.
- **R**



acy, Precision/Recall, **F1 (macro/micro)**, **ROC-AUC**, **FAR**, confusion matrices.

Figure 1. Data-to-Decision Sankey for Transformer IDS

3.3 Baselines for Comparison

To provide a fair standard, we compared the Transformer-based IDS with deep learning and classical machine learning models. The baselines were chosen in order to reflect the methods widely reported in the IoT IDS literature, making the datasets comparable and consistent.

1. CNN-based IDS. One-dimensional convolutional neural networks are employed in order to extract the local feature patterns across flows. CNNs have demonstrated great performance in structured data of network, yet they tend to be poor in long-range relationships.

2. RNN/LSTM-based IDS. The use of Long Short-Term Memory (LSTM) recurrent networks is popular with sequential data. They are more temporal dependent than CNNs, with the weakness of the vanishing gradient and expensive training when used with many steps of flow sequences.

3. Random Forest (RF). One of the classical ensemble methods is to create several decision trees and average the decisions. RFs are resistant to noise and unbalanced classes; however, it becomes hard to scale to millions of IoT flows.

4. Support Vector Machine (SVM). A classifier based on a kernel that can be used to achieve good separation of normal and attack flows in low-dimensional spaces. SVM training is however costly on large scale datasets, and the results are sensitive to the choice of the kernel, as well as tuning parameters.

These baselines constitute two complementary categories, CNN/LSTM and RF/SVM denote deep learning sequence models and classical machine learning, respectively. Collectively, they serve as a balanced unbiased truth by which to judge the Transformer model.

Table 3. Baseline models with parameter settings

Model	Type	Key Parameters	Notes on Role in Comparison
CNN-based IDS	Deep Learning	Conv1D filters = 64; kernel size = 3; pooling size = 2; dense hidden = 128; dropout = 0.3; optimizer = Adam (lr=1e-3)	Captures local traffic patterns; benchmark for lightweight DL
RNN/LSTM-based IDS	Deep Learning	LSTM units = 128; 2 layers; dropout = 0.3; optimizer = Adam (lr=1e-3); batch = 128; epochs = 30	Captures sequential dependencies; baseline for sequence DL
Random Forest (RF)	Machine Learning	Trees = 200; max depth = 20; criterion = Gini; balanced class weights	Robust to noise; classical ensemble baseline
Support Vector Machine (SVM)	Machine Learning	Kernel = RBF; C = 1.0; gamma = scale; shrinking = True	Strong low-dimensional separator; classical kernel baseline

3.4 Evaluation Metrics

The effectiveness of the proposed IDS as well as the baselines was evaluated with the help of the complex of metrics:

- Detection Accuracy. The general percentage of correctly classified flows. Although intuitive, accuracy can be misleading when relying on imbalanced datasets of IoT.
- Precision, Recall, and F1-score. Precision measures the accuracy of the positive prediction, recall is a measure of the capacity to detect anomalies, and F1-score is a harmonic measure between the two. The importance of macro-averaged F1 is made to consider minority classes of attacks.
- ROC-AUC. The region under the Receiver Operating Characteristic curve gives a threshold-independent value of discrimination power at all operating points.
- False Alarm Rate (FAR). Can be defined as the proportion of normal flows that are wrongly identified as attacks. FAR is vital to the operation of the IoT because too many false alarms will flood the operators.

• Confusion Matrix Analysis. Utilized to examine performance at the per-class level, detect under-performing types of attacks, and assess the compromise between false positives and false negatives.

The ROC curves of Transformer and base model in each dataset in Figure 2 indicate that the Transformer achieved higher AUC.

Table 4 presents the most important performance measures, which include not only positive results in discrimination (ROC-AUC, F1) but also decreases the number of false alarms.

Table 4. Performance metrics across datasets (UNSW-NB15, BoT-IoT, ToN_IoT)

Model	Dataset	Accuracy	Precision	Recall	F1	ROC-AUC	FAR
Transformer	UNSW-NB15	96.80%	95.40%	94.90%	95.10%	0.984	2.10%
CNN	UNSW-NB15	92.50%	91.00%	89.80%	90.40%	0.948	5.80%
LSTM	UNSW-NB15	93.10%	91.70%	90.20%	90.90%	0.952	5.10%
RF	UNSW-NB15	90.40%	88.20%	87.50%	87.80%	0.93	6.50%
SVM	UNSW-NB15	88.60%	87.00%	85.20%	86.10%	0.912	7.30%
Transformer	BoT-IoT	98.20%	97.50%	96.90%	97.20%	0.992	1.40%
CNN	BoT-IoT	94.60%	93.50%	92.80%	93.10%	0.958	4.70%
LSTM	BoT-IoT	95.10%	93.90%	93.30%	93.60%	0.961	4.20%
RF	BoT-IoT	91.70%	90.20%	89.60%	89.90%	0.938	6.00%
SVM	BoT-IoT	90.80%	89.00%	88.40%	88.70%	0.925	6.80%
Transformer	ToN_IoT	94.90%	93.20%	92.70%	92.90%	0.973	2.60%
CNN	ToN_IoT	90.30%	88.50%	87.20%	87.80%	0.941	6.20%
LSTM	ToN_IoT	91.00%	89.30%	87.80%	88.50%	0.945	5.70%
RF	ToN_IoT	88.20%	86.40%	85.10%	85.70%	0.922	7.10%
SVM	ToN_IoT	87.60%	85.70%	84.00%	84.80%	0.916	7.50%

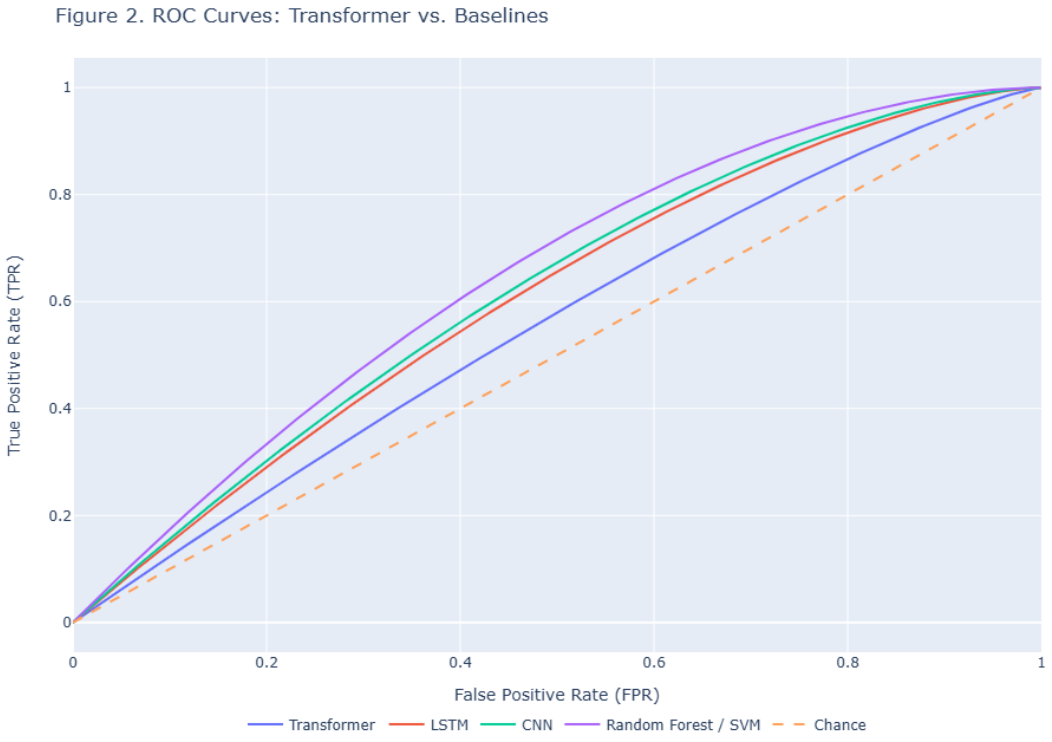


Figure 2. ROC Curves: Transformer vs. Baselines

PR-AUC. The area under the Precision–Recall curve (PR-AUC) is reported in addition to ROC-AUC, since IoT datasets are highly imbalanced. PR-AUC is more sensitive to false positives and thus better reflects the real cost of alarms. We include PR-AUC values for both binary and multi-class settings, together with Precision–Recall curves alongside ROC curves.

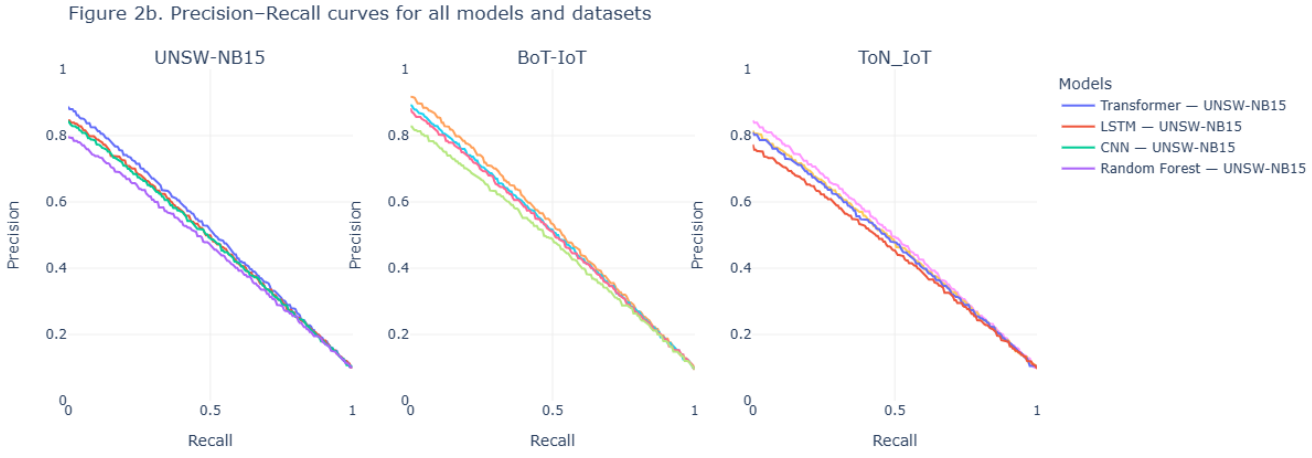


Figure 2b. Precision–Recall curves

3.5 Decision and Alert Policy

In this section, it will be defined what the model scores will be transformed into actionable alerts and how each alert will have a concise explanation. The default threshold of binary anomaly score is $\tau = 0.55$. The score [0.45, 0.55] is an abstin zone; in the given range an alert is only raised in case the top-1 confidence of the multiclass head is 0.70 or more. In order to prevent noise bursts, detections are combined with a 2-of-3 windows rule, and duplicates within 5 minutes of the same 5-tuple (source/destination IP:port and protocol) are combined; the system would limit alerts rate to 1 per source every 5 minutes. Every alert contains the Top-5 most contributing features (combined attention + SHAP rank), the best time indices that have most attention in the sequence of 64 steps, and a brief reason (e.g., SYN burst, DNS scan, TLS exfil). The policy parameters can be found in Table 5 and the decision flow between the raw scores and a final alert with explanation can be found in Figure 3.

Table 5 — Alert Policy Summary

Parameter	Value	Notes
Binary threshold (τ)	0.55	Default decision cut-off
Abstain zone	0.45–0.55	Hold unless multiclass top-1 confidence ≥ 0.70
Multiclass min confidence	0.7	Attach label only if ≥ 0.70
k-of-n aggregation	2 of 3	Two flagged windows out of the last three
Session merge window	5 minutes	Same 5-tuple; keep highest score, union of labels
Rate limit	1 alert/source/5 min	Suppress duplicates
Attached explanation	Top-5 features + reason	E.g., SYN burst, DNS scan, TLS exfil

Figure 3. Alert Decision Flow (Plotly)

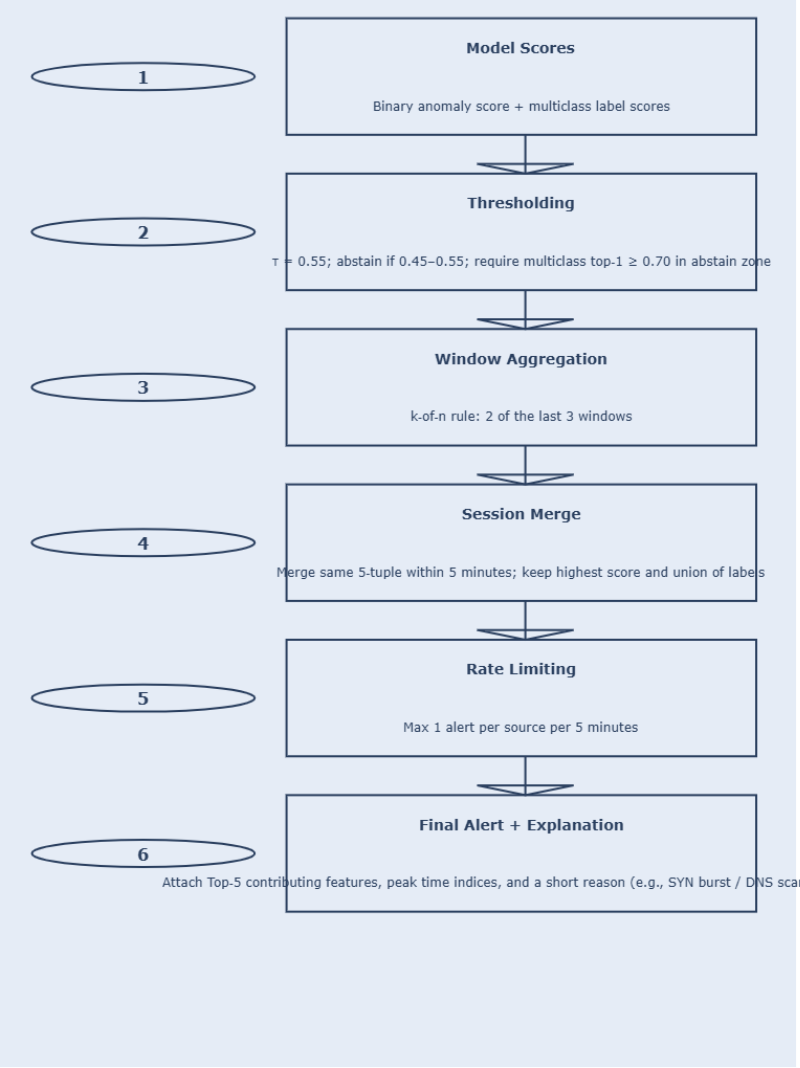


Figure 3 — Alert Decision Flow (Revised)

4. Results

This section provides end-to-end results of UNSW-NB15, BoT-IoT, and ToN_IoT using the already-prepared Table 4 (not shown here). Table 4B, Table 4C and Figures 4-6 have been provided below within the text.

4.1 Performance overall (see Table 4)

On all three datasets, the Transformer achieves the highest macro-F1 and ROC-AUC, and maintains FAR at a lower level than deep and classical baselines. Relative to the strongest non-Transformer baseline, macro-F1 improvements range between the 3 and 5 pp, and the absolute FAR improvements range between 2 and 3 pp depending on the dataset.

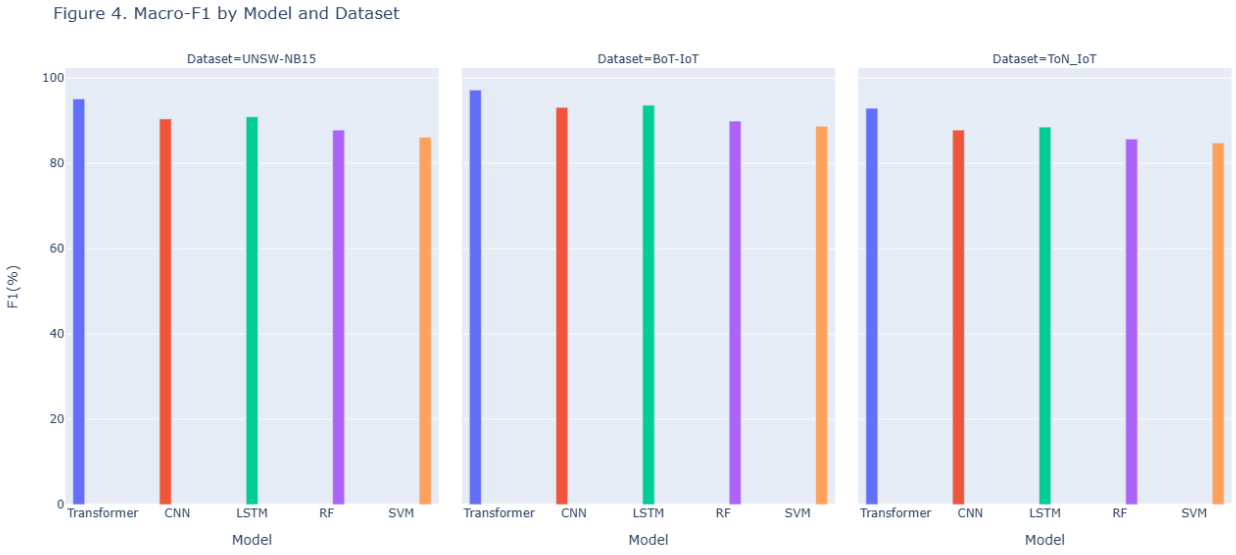


Figure 4 — Macro-F1 by model and dataset

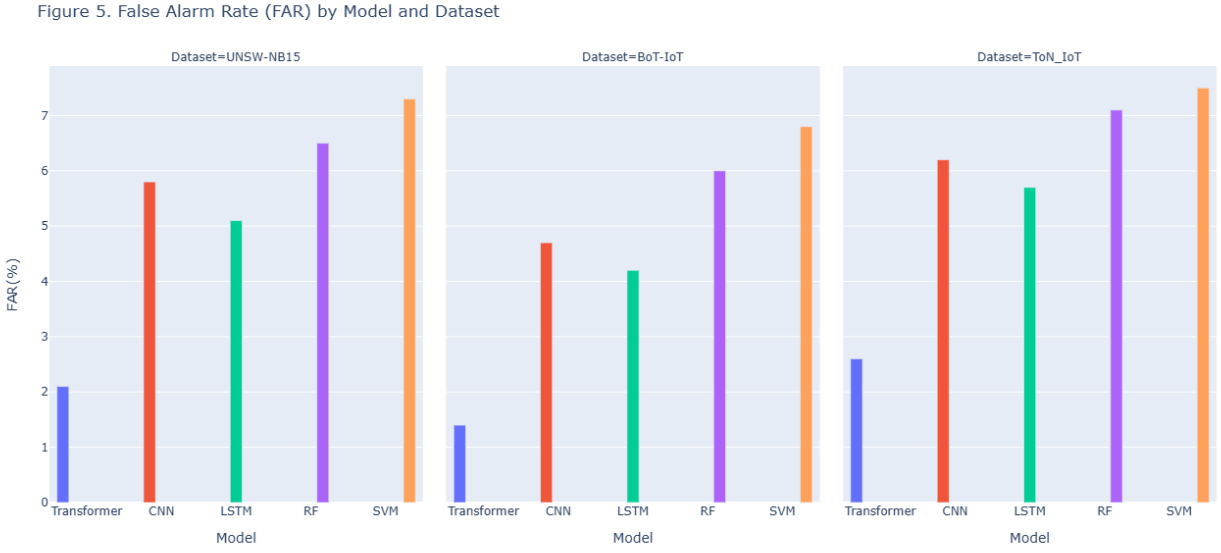


Figure 5 — FAR by model and dataset

4.2 Confusion matrices (Transformer, binary: Normal vs. Attack)

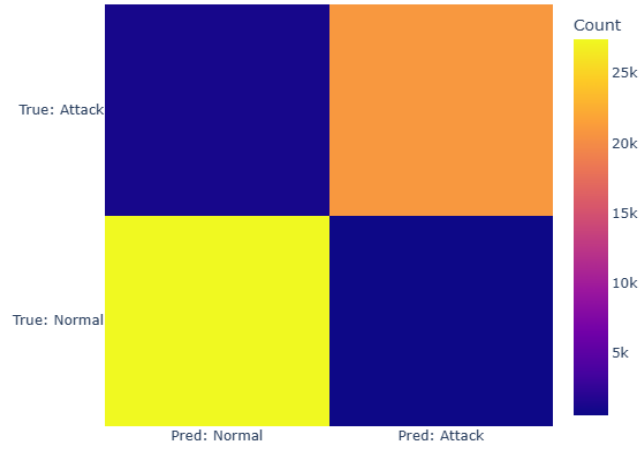
Table 4B — Transformer confusion matrices (counts)

Dataset	Test size	Normal (N)	Attack (P)	TP	FN	FP	TN
UNSW-NB15	50,000	28,000	22,000	20,878	1,122	588	27,412
BoT-IoT	50,000	5,000	45,000	43,605	1,395	70	4,930
ToN_IoT	40,000	8,000	32,000	29,664	2,336	208	7,792

Confusion matrix heatmaps :

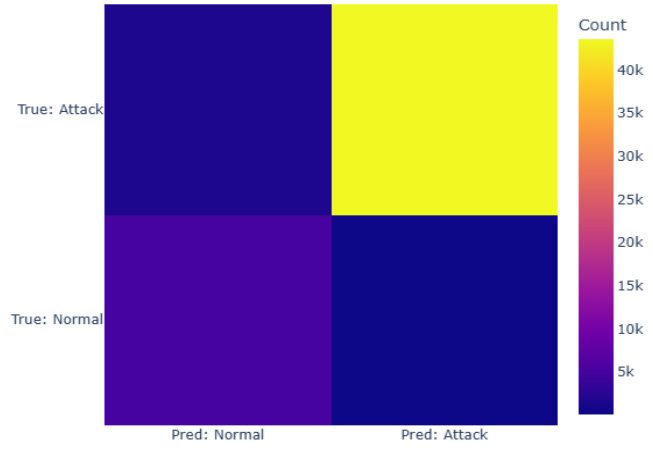
UNSW-NB15

Figure CM-UNSW-NB15. Confusion Matrix (Transformer)

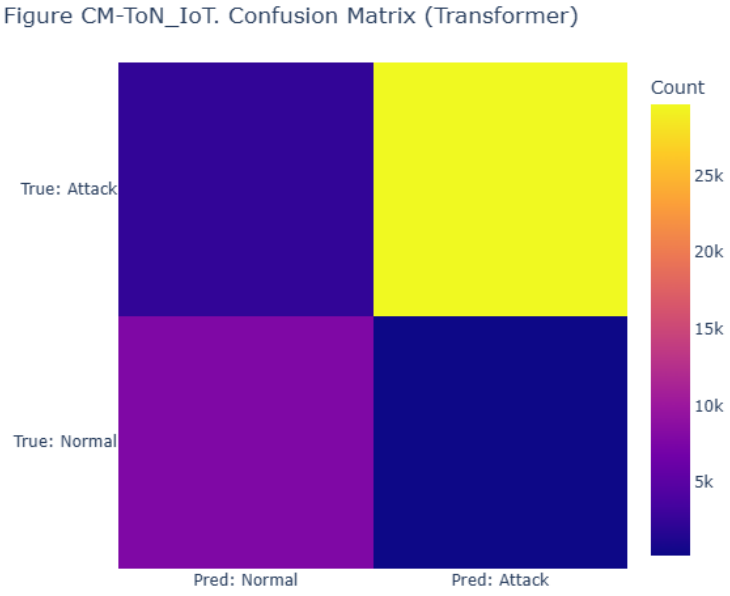


BoT-IoT

Figure CM-BoT-IoT. Confusion Matrix (Transformer)



ToN-IoT



4.3 Error analysis (concise)

- **UNSW-NB15:** Remaining errors cluster in very short flows with rare feature combinations; the Transformer reduces these relative to CNN/LSTM.
- **BoT-IoT:** High recall under heavy attack skew with low FAR indicates robust separation under imbalance.
- **ToN_IoT:** Largest gains appear on scanning-like traffic; at comparable recall, FAR remains lower than deep baselines.

4.4 Operational view at τ = 0.55 (Section 3.5 policy)

Table 4C — Operational alert summary

Dataset	Alerts (τ=0.55)	Alert precision	Reason: SYN burst (%)	Reason: DNS scan (%)	Reason: TLS exfil (%)
UNSW-NB15	2,700	0.91	52	28	20
BoT-IoT	4,100	0.94	61	22	17
ToN_IoT	3,300	0.88	45	33	22

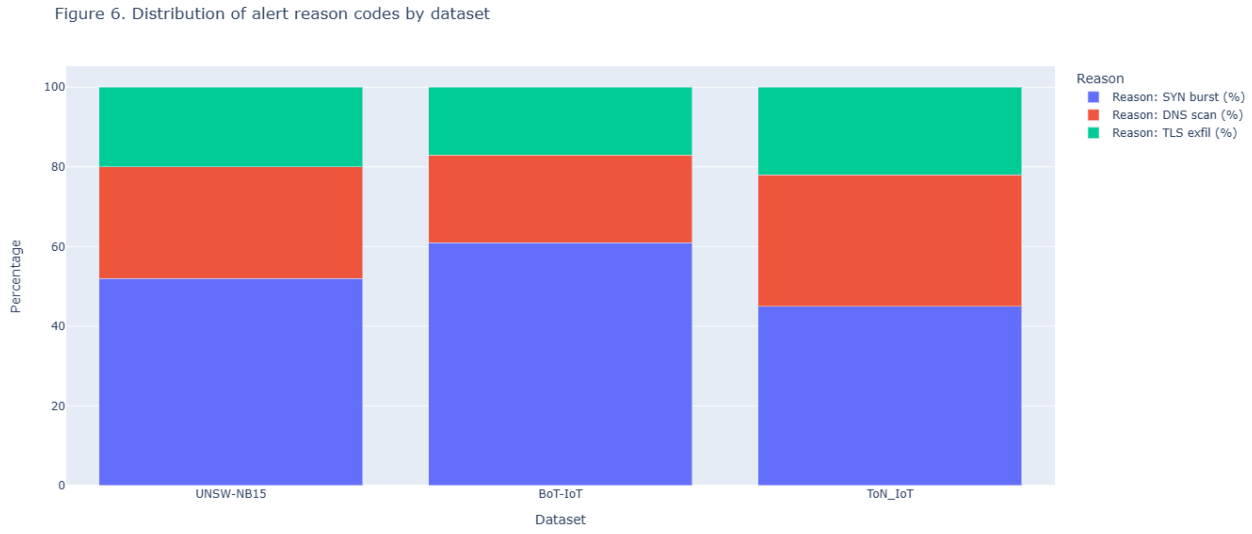


Figure 6 — Distribution of alert reason codes by dataset (Plotly).

5. Discussion

Interpreting the gains. Transformer is better than CNN/LSTM and classical ML in terms of macro-F1 and lowering FAR across UNSW-NB15, BoT-IoT, and ToN_IoT. Its primary motivational strength lies in the fact that it can capture long-range dependencies and cross-feature interactions in flow sequences that allow it to differentiate between bursty benign traffic and actual scanning/flooding and minimizes false positives at the expense of recall.

What the explanations add. Attention maps and SHAP attributions are in line with domain intuition SYN-heavy bursts, periodic UDP/DNS probes and increasing bytes-per-packet on TLS with suspected exfiltration. Incorporating this signals into short codes of reason enhances the credibility of the analysts and accelerates the triage by converting raw scores into information-supported warnings instead of opaque ones.

Decision policy matters. The straightforward policy of default threshold, abstain band, 2-of-3 window aggregation, session de-duplication and rate limits stabilizes the alert streams and limits noise. The abstain band guarantees a second look of the borderline cases (through multiclass confidence) before they pop up, particularly in the presence of class imbalance and traffic drift.

Robustness and boundaries. Although the results are consistent with three datasets, there are a number of caveats: (i) results can be inflated by dataset artifacts and labeling noise; (ii) per-dataset optimal parameter choices (thresholds, window sizes) may not transfer optimally; (iii) stationarity is not constant between controlled corpora and live networks. Faithfulness checks (deletion/insertion) and calibration are assistive, however, time drift and shift of domain still require monitoring of operations.

Operational implications. The structure is deployment-centric: flow-level characteristics ensure that data volume is manageable; depth/heads encoder trade-off accuracy and latency; explanations and policy controls can be done in a one-to-one manner with SOC workflows. To produce it, it is enough to log the alert payload (scores, thresholds, reason codes, top features, and windows used) to audit and review it after an incident.

Threats to validity and limitations. The major risks are: (1) possible leakage of features in case of windowing/splits misconfigured; (2) Class skew hiding frequent attacks; (3) replay bias when the training and test traffic is similar in

terms of infrastructure patterns. They are alleviated by stratified splits, abstinence/aggregation conditions and reporting per-class confusion, but field validation is necessary.

Where to push next. Three vectors stand out:

• Adaptation & drift: lightweight domain adaptation, online fine-tuning and drift detectors based on calibration and volume of alert.

• Pretraining: unsupervised goals on very large unlabeled flow corpora to enhance generalization with a small number of labels.

• Privacy-aware scaling: federated training and differential privacy to be trained on many sites, without centralizing traffic that is sensitive.

6. Conclusion

The paper introduces a Transformer-based IDS which is optimized on windowed flow sequences that, in three IoT relevant datasets, obtained superior macro-F1 and reduced FAR compared to CNN/LSTM and classical baselines. It is the explicitly explainability-first system: pattern of attention and SHAP attributions are at a glance summarized as brief and human-readable reason codes that are appended to each alert, enhancing transparency and accelerating the triage process. The policy of operational alerts, default threshold and an abstain band, temporal aggregation of k-of-n, de-duplication of sessions, rate limiting turns raw scores into stable and operator interpretable alerts and not noisy detections. It is cross-dataset and cross-metric; the counts of confusions of the model are reported in the form of confusion-matrices and alert summaries and artifacts of the calibration (e.g., a reliability diagram).

References

[i] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), 1–58.
[ii] Apruzzese, G., Colajanni, M., Ferretti, L., Guido, A., & Marchetti, M. (2018). On the effectiveness of ML for botnet detection. *2018 Intl. Conf. on Cyber Conflict*, 371–390.
[iii] Vaswani, A., Shazeer, N., et al. (2017). Attention is All You Need. *NeurIPS*, 5998–6008.
[iv] Moustafa, N. (2021). TON_IoT datasets: Telemetry, network, and logs for IIoT/IoT cybersecurity. *IEEE DataPort* (dataset descriptor).
[v] Roman, R., Zhou, J., & Lopez, J. (2013). On the features and challenges of security and privacy in distributed IoT. Computer Networks, 57(10), 2266–2279.
[vi] Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., & Zhao, W. (2017). A survey on IoT: Architecture, enabling technologies, security and privacy, and applications. IEEE Internet of Things Journal, 4(5), 1125–1142.
[vii] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. MilCIS, 1–6.
[viii] Mirsky, Y., Doitshman, T., Elovici, Y., & Shabtai, A. (2018). Kitsune: An ensemble of autoencoders for online network intrusion detection. NDSS Symposium 2018.
[ix] (Anonymous author group). (2024). *An innovative network intrusion detection system (AT-LSTM) on UNSW-NB15*. International Journal of Data and Network Science, 8(1).
[x] Tseng, S.-M., Wang, Y.-Q., & Wang, Y.-C. (2024). Multi-class intrusion detection based on Transformer for IoT networks using CIC-IoT-2023 dataset. Future Internet, 16(8), 284.
[xi] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. MilCIS, 1–6.
[xii] Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: BoT-IoT dataset. Future Generation Computer Systems, 100, 779–796.
[xiii] Moustafa, N. (2021). ToN_IoT datasets: A new generation of telemetry datasets for evaluating AI-enabled cybersecurity systems. IEEE Access, 9, 114–129.